●ホビー・エレクトロニクスの情報誌 昭和52年9月1日発行(毎月1回1日発行)第2巻第9号(通巻11号)
昭和52年1月11日 第三種郵便物認可

Microcomputer

TV Game

Music Synthesizer

Laser Art

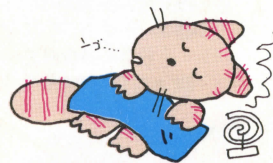
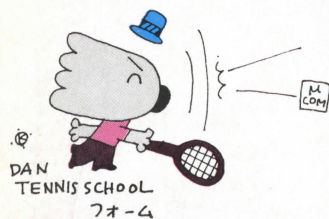
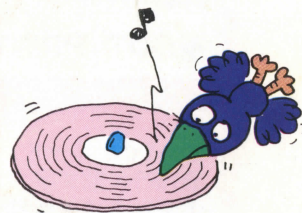
ついに登場!
衝撃 第2弾!
★6800 4K BASIC
★8080 2K BASIC
★8080スロットマシン・ゲーム
★★レコード+全リスト付!

特集 BASICを使いこなそう!

キャラクタ・ディスプレイを使いこなそう

シンセサイザ リズムボックスをつくろう

S-100BUSを斬る!

新連載 8080マイコンの基礎と製作

特価—350 yen

リースバック端末・ミニコン・周辺機器続々入荷中

ミニコン系統

（I/O付システム）一以下は一例です一

品名	価格
FACOM RE システム・I/O・ラインプリンター付	¥1,300,000
HITAC 10 ASR-33・(大量入荷)	¥ ＊＊＊＊＊
HITAC 10 II/A(OEM価格にて販売中)	¥ ＊＊＊＊＊
WANG 720B(磁気ディスク・IBMコンソール付)	¥1,350,000
MELCOM83 STANDARD フルセット	¥ 580,000
MELCOM83 DELUXE フルセット	¥ 610,000
DEC PDP8E	¥1,400,000
オリベッティP603(MLU付) フルセット	¥ 400,000
リコム8	¥ 500,000
リコム6(56時間使用一新品)システム	¥ 850,000
NEAC1240システム	¥ 680,000
リコー416B・416C	¥ 250,000
FACOM230－10	¥1,200,000
USAC－720－10(高速PTP－PTR付131KW)	¥ 850,000
TOSBAC 1500/20フルシステム	¥2,500,000
HITAC 8300フルシステム	¥2,350,000
ザイネテックス1100(NOVA＋XYプロッター他)	¥4,500,000
OKITAC 4300 フルシステム	¥1,600,000

デスクトップ型

品名	価格
HP9100A 磁気カードプログラム・プリンター付	¥ 160,000
HITAC MINI	¥ 220,000
オリベッティP101(DELAY LINE 1K)	¥ 90,000
セイコー S301	¥ 115,000
リコータイパック16B(MCT付)	¥ 120,000
オリベッティロゴス250・320	¥ 10,000

端末機系統

（他多数在庫）

品名	価格
テレタイプ社ASR－33 (ASCII) 新同様	¥ 350,000
オリベッティTE308-318(ASCII)	¥ 180,000
リコータイパースタンダード(4種)他200・600型	¥
ブラシ×1（リーダー装備数）	¥ 85,000
ブラシ×2（リーダー装備数）	¥ 90,000
フォト×1（リーダー装備数）	¥ 95,000
フォト×2（リーダー装備数）	¥ 100,000
オキタイパー6000	¥ 120,000
富士通DR7300	¥ 180,000
谷村PTS1000	¥ 100,000
NEAC・G-201(新品同様)IBM使用	¥ 220,000
谷村SKS100	¥ 40,000
岩通ターミナル2020	¥ 130,000
サイバーコムKEY TOCASSET	¥ 60,000
サイバーコムCASSET TOMT	¥ 150,000
IBM029カードパンチャー	¥ 150,000
IBM029カードパンチャーP付	¥ 300,000
リコーテープパンチャー	¥ 19,000
リコーテープリーダー(ブラシ)	¥ 14,000
リコーテープリーダー(フォト)	¥ 19,000
IBM I/Oタイプライター	¥ 65,000
バリパンチ(電動小型カードパンチャー)	¥ 80,000
富士通テープパンチャー(6/8bit)	¥ 21,000
富士通テープリーダー	¥ 18,000

マイコンコーナー

（KIT・完成品）以下は一例です。

品名	価格
ALTAIR 680B (キット)	¥ 235,000
ALTAIR 680B	¥ 320,000
ALTAIR 8800A (キット)	¥ 285,000
ALTAIR 8800A	¥ 390,000
APPLE-I	¥ 予定価格 300,000
BIG-ONE 90L	¥ 79,000
BIG-ONE 91L	¥ 76,800
FAIRCHILD F8S(完成品)開発用モジュール	¥ 325,000
FAIRCHILD F8(キット)	¥ 62,000
FAIRCHILD F8CMM(完成品)	¥ 64,900
INTEL SDK-80	¥ 83,000
INTEL SBC-80/10	¥ 180,000
INTEL SBC-80/20	¥ 326,000
INTERCEPT JR	¥ 140,000
INTERSIL 6100	¥ 160,000
(CMOS FAMI SAMPLE)チップのみ	¥ 25,000
IMSAI 8080	¥ 336,000
JOLT(キット)(MOSテクノロジー)	¥ 63,000
KIM-1(完成品)(MOSテクノロジー)	¥ 119,000
MB2102 CPUボード(富士通)	¥ 160,000
MEK 6800D II A(モトローラ)	¥ 79,000
MICRO 8/16-80/10	¥ 79,200
NS SC/MP	¥ 39,500
TOSHIBA TLCS-12A EX-0(キット)	¥ 90,000
TOSHIBA TLCS-12A EX-1A(完成品)	¥ 498,000
TK-80	¥ 89,500
UD990 I/O	¥ 99,000

MOSTEK Z—80 好評発売中

富士通ディバイス

区分	品名	構成	構造	機能	備考	相当品	外形	価格
CPU	MB8861	8Bit	N-ch	Processor	20ns	MC6800	Dip-40	@¥ 9,000
ROM	MB7052	256×4	Bipolar	P-ROM	60ns	IM5623	Dip-16	@¥ 1,500
	MB8513	256×8	P-ch	E-P-ROM	1,000ns	I-1702A	Dip-24	@¥ 3,900
	MB8518	1024×8	N-ch	E-P-ROM	450ns	I-2708	Dip-24	@¥12,000
RAM	MB8101	256×4	〃	static RAM	〃	I-2101	Dip-22	—
	MB8111	256×4	〃	〃	〃	I-2111	Dip-18	@¥ 1,000
	MB8102	1004×1	〃	〃	〃	I-2002	Dip-16	@¥ 850
	MB8107	4090×1	〃	Dynamic RAM	300ns	I-2107	Dip-22	@¥ 2,200
	MB8224	〃	〃	〃	280ns	I-2104	Dip-16	@¥ 2,200
	MB8862		〃	peripheral interface Adapter		MC-6820	Dip-40	@¥ 4,200
	MB8863		〃	A・C・I・A		MC-6850	Dip-24	@¥ 5,000
	MB8867		Bipotar	cloch Geueretar		—	Dip-24	@¥ 3,800
	MB8868		N-ch	Transmitter/Receiuer		WP1602A	Dip-40	@¥ 5,000
	MB 424	4Bit	Bipolar	Bus Driuer/Receiuer		8T26	Dip-16	@¥ 950
	MB 425	〃	〃	Bus Driuer Non Inreerting		I-8216	Dip-16	@¥ 950
	MB 426	〃	〃	〃 Inverting		I-8226	Dip-16	@¥ 950
	MB427p	〃	〃	Clock Driuer		SN75113	Dip-16	@¥ 950
	MB 471	8Bit	〃	Input/output port		I-8212	Dip-24	@¥ 1,200

通信販売を行なっておりますので御利用下さい

◆DOT PRINTER

＊5×7BIT ＊75行/分 ＊40キャラFIFOメモリ
＊40キャラ/行 ＊64文字・ASCII ＊パラレル入力

¥150,000(キット)

CRT DISPLAY VT-1020/II

- ●ASCIIコード ●5×7 DOTマトリックス
- ●16行×32文字×2ページ（増設時6ページ）
- ●テレタイプ仕様インターフェイス
- ●110ボー～300ボー可変ボリューム付・PTP・PTRへの接続容易
- ●CLOCK内蔵 ●シリアル入力（パラレル可）
- ●オーディオカセットインターフェイス付(300まで可)
- ●内部編集機能付 ●各種オプション取付可
- ●カーソルコントロール(6機能) ●家庭用TVに接続可能です
  （GO HOME・バックスペース・ラインフィード・バックライン等）
- ●ASR－33コンパチブル（20mmA カレントループ）

¥183,000(本体価格)

ショールーム

㈱アスターインターナショナル

〒160 東京都新宿区新宿1－1－11 武シートビル1F・5F
（新宿1丁目バス停前）

☎東京03-354-2661・2662・2663(代表)

CONTENTS 1977年9月号

特集•••BASICを使いこなそう❢

＊イラスト＝はらJIN＋きむらしんじ

HOBBY ELECTRONICS JOURNAL

I/O

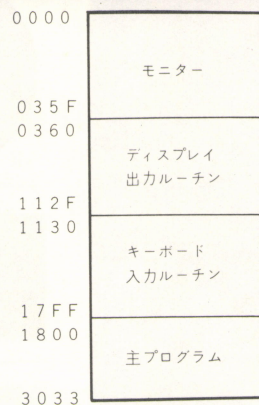
らんだむ・あくせす・でくしょなり
Random Access Dictionary

●メモリー・マップ

その名の通りメモリーの地図のことで,メモリー内の各サブ·プログラムや,データが,何番地から何番地に入っているかを示したもの.

1つの適用プログラムは普通は,いくつかのサブ・プログラムを組合せて構成される.

たとえばキーボードからのデータを読んで何らかの計算を行い,テレビディスプレイに表示するプログラムを考えると,このプログラム中には,計算を行うだけでなく,キーボード入力プログラム,テレビ表示プログラムが必要になる.

というのは,キーボードの入力プログラムなどは他の目的にも使えるので,個々に作成せずに,サブ・プログラムとして独立に作成するのが普通だからで,その場合,各サブ・プログラムをメモリー上のどこに配置するか,したかを示すのがメモリーマップである.

●フラグ

コンピュータに計算させるデータではなく,その過程に必要な演算の結果(正負など)や,I/Oなどの状態を表す情報.命令でこれを調べ,条件分岐命令によって異った処理を行わせるために用いられる.

マイコンに限らずコンピュータで演算を行うと,その結果が正か0か負かキャリー,ボロー(最上位の桁の加減算の桁上りや借り),オーバーフロー(桁あふれ)が起ったかなどをCPUが検出し,それを演算フラグと呼ぶレジスターに入れる.

条件分岐命令とは,このフラグの内容が条件に合致したなら分岐を行い,そうでない時は,次の命令を実行する.

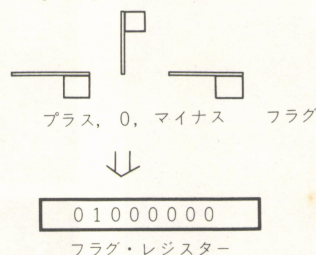
コンピュータの処理とは,与えられたデータをプログラム(命令の集り)で処理し,何らかの出力装置へ出力していくものである.

つまり処理に使うデータや入力されたデータがあり,これらを一連の命令によって処理していくが,この中間での判断を行う情報がフラグである.

たとえば,2次方程式の実根の個数は,根の公式によってルートの中の正,0,負の3通りに分けられる.

したがって根の個数を求めるには,

$$b^2-4a\cdot c$$

を計算し,演算フラグをセットし,条件分岐命令を正の時,0の時,負の時に分岐させ,分岐先におのおの根の個数を出力する命令を書けばよい.

●インダイレクト・アドレッシング

間接アドレッシングとも言い,演算命令や,分岐命令において,命令のアドレス部の内容が対象となるデータや分岐先ではなく,その内容が再びアドレスを示す方式.

たとえば

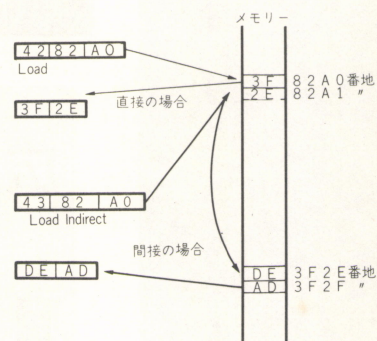
4282A0(16進)

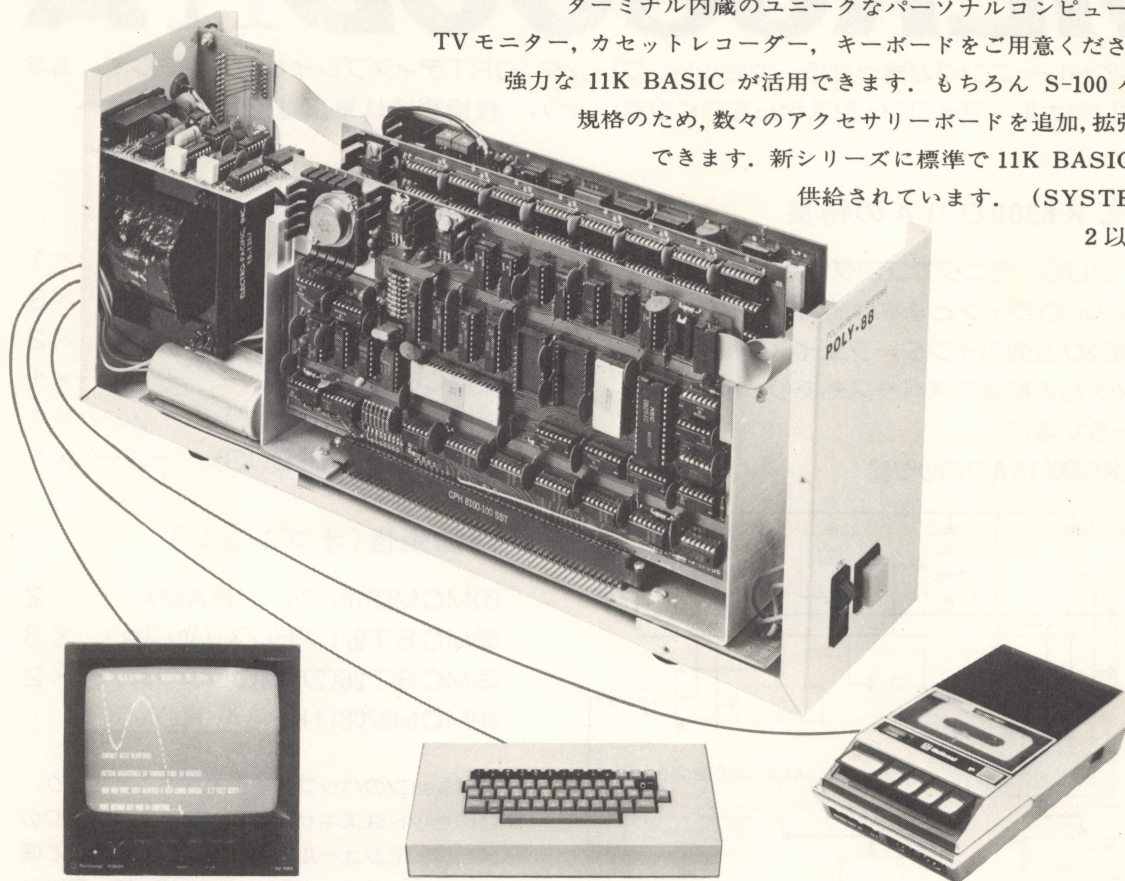
という命令が,82A0番地の内容を(ただし,2バイト)レジスターにLoadする命令だとする.もし82A0番地の内容が3F2Eだとすると,レジスターの内容は,3F2Eになる.ところが

4382A0(16進)

という命令が82A0番地の内容を間接で2バイトLoadする命令だとする.

この場合は82A0番地の内容は3F2Eなので,本当のアドレスは,82A0の内容つまり3F2E番地にある.

ここで3F2E番地の内容がDEADなら,レジスターにはDEAD(16進)が入る.

The POLY 88 Microcomputer System

ターミナル内蔵のユニークなパーソナルコンピュータ.
TVモニター，カセットレコーダー，キーボードをご用意ください.
強力な 11K BASIC が活用できます．もちろん S-100 バス規格のため，数々のアクセサリーボードを追加，拡張ができます．新シリーズに標準で 11K BASIC が供給されています．（SYSTEM-2以上）

●SYSTEM-2の内容／ KIT価格 ￥317,000

CPUボード，VIDEOインターフェースボード，カセットインターフェースボード，キャビネット，バックプレーン，電源，11K BASIC，アセンブラー

＊SYSTEM-2 でオペレーションするためには次のものが必要です．
S-100BUS メモリーボード 16Kバイト以上，TVモニター，カセットレコーダー，キーボード（ASCII）

●SYSTEM-6の内容 ／ KIT価格 ￥682,000

CPUボード，VIDEOインターフェースボード，カセットインターフェースボード，16K RAMボード，キャビネット，バックプレーン，電源，ファン，キーボード，11K BASIC，アセンブラー

＊SYSTEM-6 でオペレーションするためには次のものが必要です．
TVモニター，カセットレコーダー

11K BASICの紹介

▶仕様

・サイズ ： 11K BYTE
・コマンド ： RUM, LIST, SCR, CLEAR, REN, CONT, SAVE, LOAD, VERIFY
・ステートメント ： LET, IF, THEN, ELSE, FOR, NEXT, GOTO, ON, EXIT, STOP, END, REM, READ, DATA, RESTORE, INPUT, GOSUB RETURN, PRINT, POKE, OUT
・ファンクション ： FREE, ABS, SGN, INT, LEN, CHR$, VAL, STR$, ASC, SIN, COS, RND, LOG, TIME, WAIT, EXP, SQRT, CALL, PEEK, INP, PLOT

▶特徴

・フォーマッティド アウトプット ・ストリング ファンクション ・アレーのディメンションはメモリー容量によってのみ制限 ・マルチプルステートメント／ライン・リナンバー可能・IF-THEN-ELSE・INPUT "×××" × ・SAVE, LOAD, VERIFYコマンドにより，オーディオカセットをテープをファイルストレージとして使用できます。・メモリー，I/Oポートとのコミュニケート ・ビデオディスプレー上へのプロットができます。・リアルタイムクロックを利用できます。

11K BASICを使用したソフトウェアーライブラリーが到着しました。

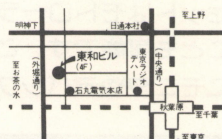
第1弾 **GAME TAPE 1** ￥6,700
LANDER, STOCK, REVERSE, HAMURABI
GAME TAPE 2 ￥6,700
BACKGAMMON, HANGMAN, WUMPUS, CRAPS
CASHFLOW

現在、当社でデモンストレーション中!!

PolyMorphic Systems 日本代理店

BYTE SHOP
the affordable computer store

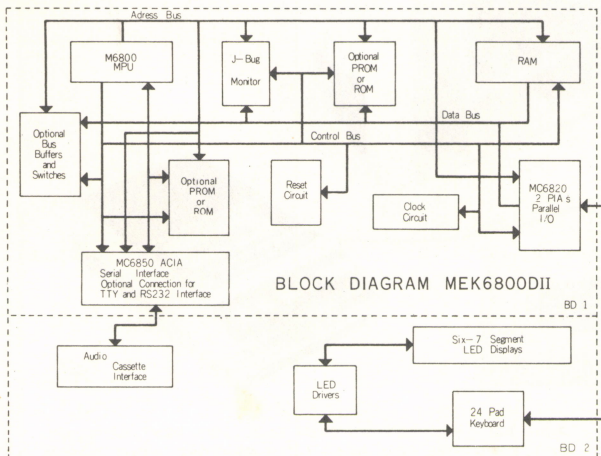
(株)バイトショップソーゴー

〒101 東京都千代田区外神田1-5-9 東和ビル4F
TEL 03(255)1984 営業時間 10：00〜7：00

MEK6800DⅡA

キットでトレーニングが終ったら、カセット、プリンタ、CRTディスプレイなどを接続して、あなたのオリジナル・マイコン・システムをつくりましょう。 技術相談は東京電子科学機材㈱へ

■MEK6800DⅡAの特長

- "J-BUG" モニタでプログラムが簡単に出来ます。
- 16本のI/Oラインと4本の制御ライン
- 直列及び並列のインターフェイス機能
- システムの拡張用スペースもあります。
- 単一5V電源

■MEK6800DⅡAブロック図

■9チップ構成

1. MC6800(MPU)……………………×1
2. MCM6810(1K RAM)…………×3
3. MC6820(PIA) ……………………×2
4. MC6850(ACIA)…………………×1
5. MC6871(CLOCK)………………×1
6. MCM6830(J-BUG)………………×1

■拡張性(オプション)

①MCM6810(128×8 RAM)………×2
②MC8T97(アドレス・バッファ)…×3
③MC8T26(2方向性バッファ)….×2
④MCM2708(1K×8A ROM)

オプションのバッファを装着することにより、このセットはエキササイザー用I/O及び諸々のメモリ・モジュールをこのセットに組合せて使えます。ワイヤラップ・エリアもバッファ用に用意されています。16ピンDIPパッケージも20個まで装着できます。

■ファームウエア

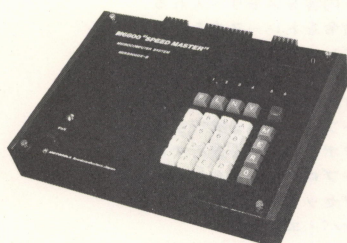
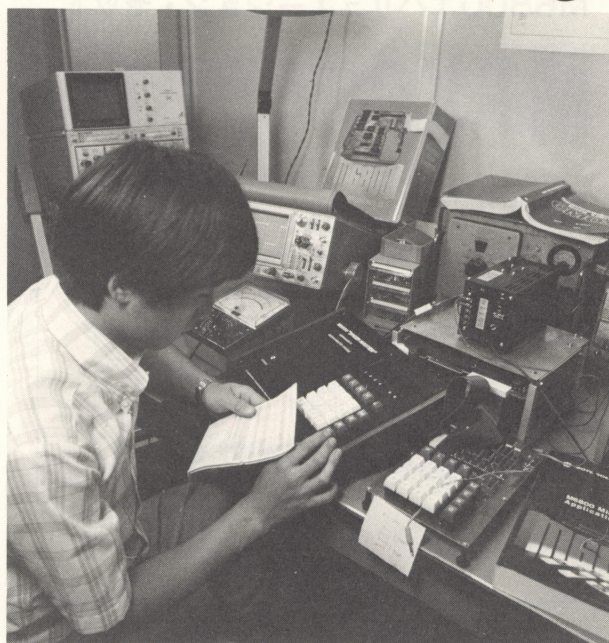
"J-BUG" モニタの機能は16進のキーボードとディスプレイ・モジュールを使ってM6800マイクロコンピュータをコントロールし、通信することも可能です。システム・キーボードは24キーで次の機能を備えています。

1. メモリ内容をカセットへ入れる
2. カセットの内容をメモリへ入れる
3. 1つの命令をトレースする
4. 5つのブレイクポイントを設定できる
5. メモリ内容を表示及びチェンジする
6. レジスタ内容を表示及びチェンジする
7. ユーザープログラムを実行する
8. ブレイクポイントから進行する
9. ユーザプログラムをアボードする
10. 相対オフセットを計算する
11. 16進法ナンバ・エントリ

このセットはモトローラ・MINIBUGⅡ、又はⅢモニタROMを("J-BUG"の代りとして) 装着することも可能です。この場合にはTTYターミナル等の直列非同期の端末を用いて "J-BUG" 同様にモニタやデバッグ等の動作を行う事が出来ます。

あなたのお手持のマイコン又は改造マイコンを使った

システム・アイデア 買います。

MEK6800DIIA
¥79,000

MEK6800DIIB
¥93,000

■申込み方法

作品には参考資料（例えば、写真、フローチャート、リスト、回路図をつけて、①住所、②氏名、③年令、④会社の業種とあなたの仕事、⑤学生の場合は、学校名・学部・回生を明記して下さい。
尚、未発表の作品に限ります。
マイコンの技術相談も受付けておりますので、ハガキまたは手紙でどしどしお寄せ下さい。11月号より相談コーナを1頁もうけましたので、あらゆる相談・質問にお答え致します。
送り先　〒101 東京都千代田区外神田2−4−4 ☎03(255)8828㈹ 東京電子科学機材㈱企画室宛

特別優秀作	1名	MEK6800DIIA・プリンタ・賞金10万円
優秀作	3名	MEK6800DIIA
アイデア作	10名	プリンタ

尚MEK6800についての意見（長所・短所・改良点）もどしどしおよせ下さい。参考意見として取り上げましたら謝礼をさし上げます。
※発表はI/O11月号に掲載致します。尚、8月号の広告に価格のミスがありましたのでおわびと訂正をいたします。 MEK6800DIIA…¥79,000、MEK6800DIIB(スピードマスター)…¥93,000
尚、採用の作品の工業所有権は東京電子科学機材の所属と致します。

モトローラ社製品についてのあらゆる相談も是非どうぞ
〈販売代理店〉
TR 東京電子科学機材株式会社 東京都千代田区外神田2-4-4 〒101 TEL.03(255)8828(代)

販売代理店　東京電子科学機材㈱長野営業所　長野県岡谷市幸町6−11　五十川ビル☎（02662）3−1074

マイコンを作る。

●入門者からプロまで使える.
強力なファームウェアと容易な拡張性

M6800エバリュエーション・セット

MEK6800DIIA

ボード状完成品

■ＭＥＫ6800ＤⅡブロック図

即納 可能です.

¥79,000

■ファームウェア

〝J-BUG〞モニタの機能はユーザーが16進のキーボードとディスプレイモジュールを使って，M6800マイクロコンピュータをコントロールし，通信することを可能にします。

システム・キーボードは，24キーで，次の機能を備えています。

1. メモリ内容をカセットへ入れる
2. カセット内容をメモリへ入れる
3. 1つの命令をトレースする
4. 5つのブレークポイントを設定できる
5. メモリ内容を表示及びチェンジする
6. レジスタ内容を表示及びチェンジする
7. ユーザープログラムを実行する
8. ブレークポイントから進行する
9. ユーザープログラムからアボートする
10. 相対オフセットを計算する
11. 16進ナンバ・エントリ

このキットは，モトローラMinibugⅡ又はⅢモニタROMを（〝J-BUG〞の代りとして）装着することも可能です。

この場合には，ＴＴＹターミナル等の直列非同期の端末を用いて，〝J-BUG〞と同様にモニタやデバッグ等の動作を行うことができます。

■拡張性（オプション）

このキットは，システムの拡張を容易にするためデバイスを追加できます。

[MCM6810 (128×8 RAM)×2
MC8T96(アドレス・バッファ)×3
MC8T26(二方向性バッファ) ×2] ＋ [MCM68316E (2K×8ROM)
MCM68708 (1K×8AROM)
MCM68308 (1K×8ROM)
HA7640 (512×8PROM)]

以上のうち，いずれか2個

オプションのバッファを装着することにより，このキットはエキササイザ用I/O及び諸々のメモリモジュールをこのキットに組合せて使うことができます。ワイヤラップ・エリアもバッファ用に用意されています。16ピンDIPパッケージも20個まで装着できます。

■価格

MC6800L(MPU)	¥8,600
MCM6810AP(1K RAM)	¥1,800
MC6820L(PIA)	¥4,250
MCM6830L(M-BUG)	¥5,000
MC6850L(ACIA)	¥4,250
MC6871B(CLOCK-GEN)	¥7,000
MC8T26P(BUS DRIVER)	¥1,200
MC8T96P(ADDR-BUFFER)	¥900

その他プラスチックパッケージも在庫あります。価格はお問合せください。

MC14433(AD CON) 3½DVM	¥3,550
MC1408L-6 (DA·CON 8bit)	¥3,950
7 (DA·CON 8bit)	¥4,950
8 (DA·CON 8bit)	¥5,950

スイッチングレギュレータ用コントロールリニアIC
MC3420P ¥2,500

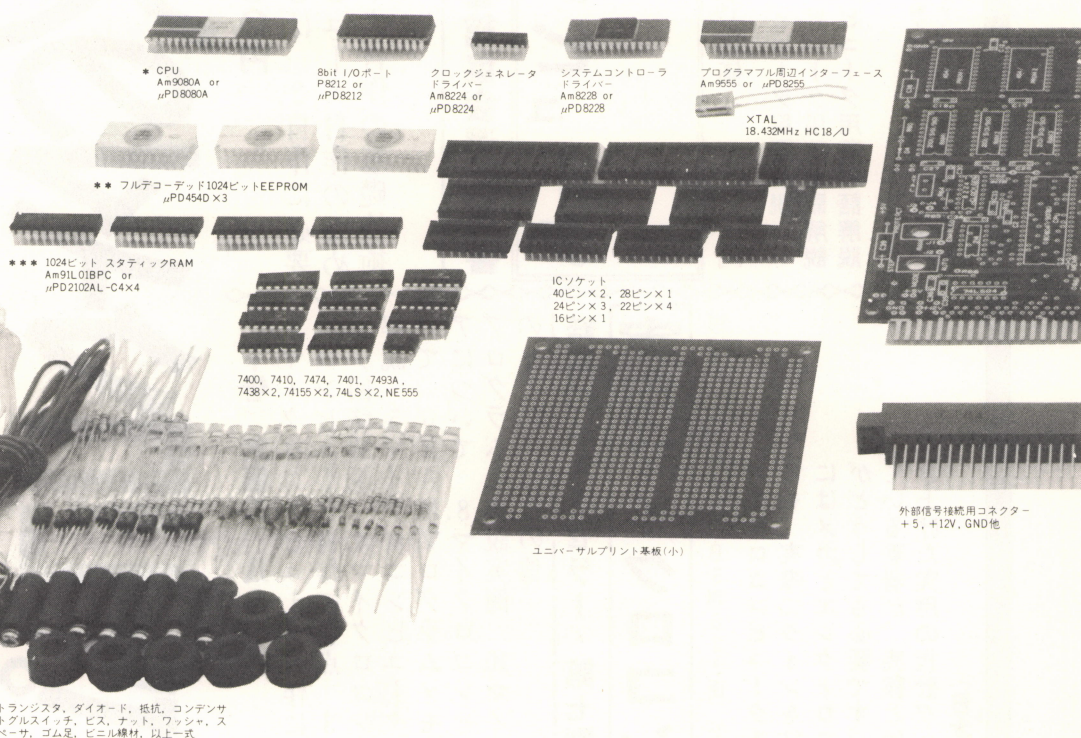
NEC TK-80. 東芝TLC-12A
EX-0の在庫もございます。

株式会社キョードー
■本　社　〒101 東京都千代田区外神田1－9－9(内田ビル3Ｆ)
経理・通販☎ 03(253)9531
■森ビル営業所　〒101 東京都千代田区外神田1－10－11(森ビル1Ｆ)
☎ 03(255)1751(代表)　☎ 03(255)1753(集積回路)
■東京ラジオデパート営業所　〒101 東京都千代田区外神田1－10－11(東京ラジオデパート1Ｆ)
☎ 03(255)1752(東芝半導体)

衝撃のマイクロコンピュータ

熱い期待に応えて—
噂の“1702A搭載
(PROM)

MK-80A

——無限の可能性を秘めた最も お求め易い価格のマイクロコンピュータ——

■MK-80AはTK-80と同一機能機を廉価にお届けすべく願いを込めて開発されたマイクロコンピュータキットの決定版です。既に大学, 企業, マニアの方々にて御使用頂いており好評を博しております。

■MK-80Aの価格¥68,000〒1,000はマニュアルを含めた価格です。なお, MK-80Aお求めの方には参考資料としてTK-80マニュアル一式をサービス致します。

■MK-80Aには専用電源POWERFUL-80の用意がございます.¥15,000〒1,000

＊ MK-80AのCPUの標準使用はAMD：Am9080Aですが御希望により, NEC：μPD8080A使用にても御納入致します。μPD8080A使用の場合でも価格は¥68,000で同一です。

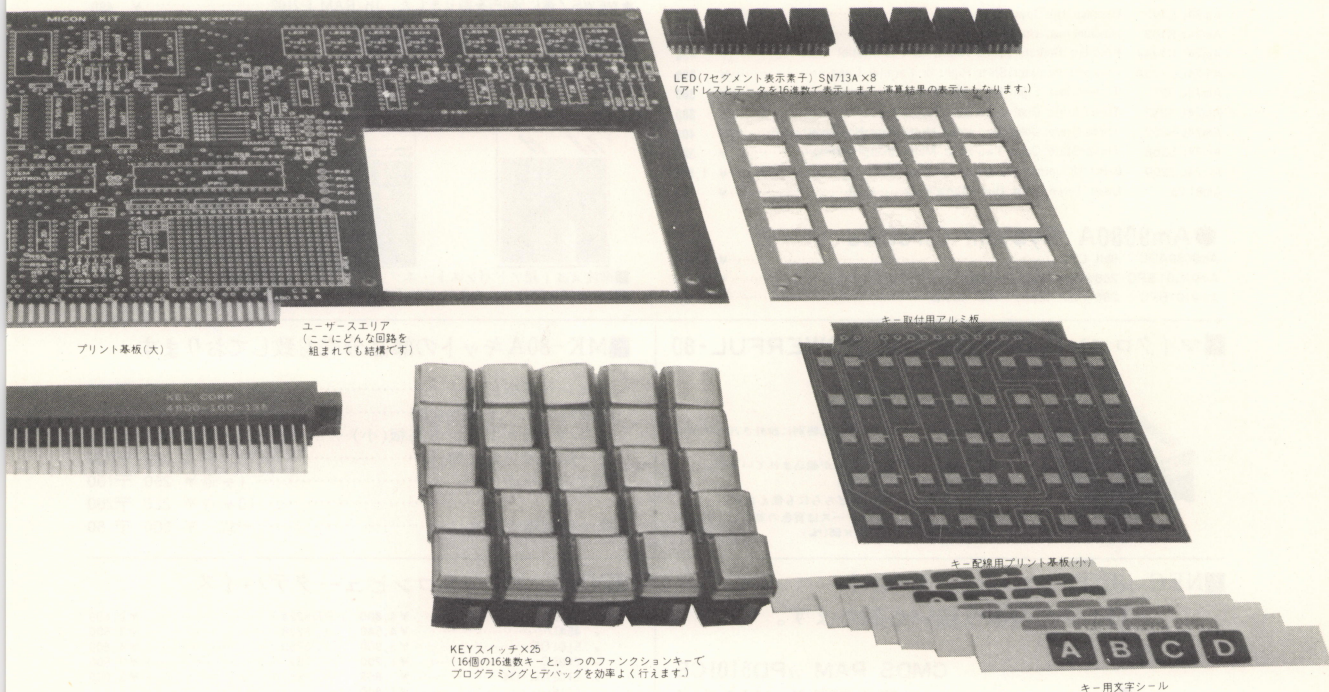
＊＊同K-80AのPROMはTK-80コンパチビリティーを保有するためにμPD454Dを使用しておりますがμPD454Dの電気的特性及び安定供給に関して問題あるため新しく1702Aを使用したMK-80Aも開発されました. 価格は¥72,000〒1,000です。1702A使用機もTK-80同一機能を保有します。又, 454D使用のMK-80A, TK-80に1702Aを使用できるようにするためのアダプターも用意されています。

＊＊＊MK-80AのRAMの標準使用はnMOSのAm91L01BPCですが御希望によりCMOSのμPD5101CE使用にて御納入致します。その際の価格は¥72,000〒1,000です。

──その全貌を遂に公開！

型”機種も販売開始！

¥68,000

- TK-80セカンド
- マニュアル共で
- 送料¥1,000

＊御注文は現金書留又は，銀行振込みにてお申込み下さい。大学，官庁関係等は所定のお支払い手続きでお求めになれます。

お求めは **AER** ADVANCED EQUIPMENT RESEARCH CORP.
I/O係 〒182 東京都調布市小島町1-5-1 ☎0424-85-7834(代)

製造元 ㈱インターナショナル・サイエンティフィック I/O係 〒193 東京都八王子市小比企町2957-9 ☎0426-25-7941(代)

Advanced

1歩進んだAMDの半導体は
より高速へ、
より少ない消費電力へ、
より小さなチップへ、
より高出力へ……と。

●amd LOW POWER SCHOTTKY TTL

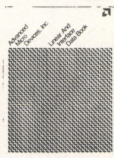
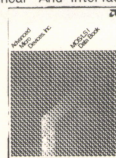
Am74LS138	One-of-Eight Decoder/Demultiplexer	¥ 370
Am74LS139	Dual One-of-Four Decoder/Demultiplexer	¥ 370
Am74LS151	Eight-Input Multiplexer	¥ 330
Am74LS153	Dual Four-Input Multiplexer	¥ 330
Am74LS157	Quad Two-Input Multiplexer;Non-Inverting	¥ 330
Am74LS158	Quad Two-Input Multiplexer;Inverting	¥ 350
Am74LS160	Synchronous BCD Decade Counter, Asynchronous Clear	¥ 550
Am74LS161	Synchronous Four-Bit Binary Counter, Asynchronous Clear	¥ 550
Am74LS162	Synchronous BCD Decade Counter, Synchronous Clear	¥ 550
Am74LS164	8-Bit Serial-In, Parallel Out Shift Register	¥ 450
Am74LS174	Six-Bit Register with Common Clear	¥ 380
Am74LS175	Quad Register with Common Clear	¥ 450
Am74LS181	Four-Bit ALU/Function Generator	¥ 1,000
Am74LS190	Synchronous BCD Decade Up-Down Counter;Single Clock	¥ 600
Am74LS191	Synchronous Four-Bit Binary Up-Down Counter;Single Clock	¥ 600
Am74LS192	Decimal Up/Down Counter	¥ 600
Am74LS193	Hexadecimal Up/Down Counter	¥ 600
Am74LS194A	Four-Bit Register;Shift Right, Left or Prallel Load	¥ 450
Am74LS195A	Four-Bit Register;Shift Right or Parallel Load	¥ 390
Am74LS251	Three-State Eight-Input Multiplexer	¥ 380
Am74LS253	Three-State Dual Four-Input Multiplexer	¥ 380
Am74LS257	Three-Stare Quad Two-Input Multiplexer;Non-Inverting	¥ 400
Am74LS258	Three-State Quad Two-Input Multiplexer;Inverting	¥ 380
Am74LS299	8-Bit Universal Shift/Storage Register	¥ 1,800
Am8T26	Quad Three-State Bus Transceiver	¥ 900

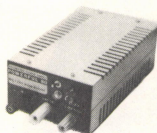
●Am9080A System Circuits

Am9080ADC	8bit CPU	¥ 4,800
Am91L01BPC	256×4bit static RAM400ns	¥ 1,200
Am9101BPC	256×4bit Static RAM 400ns	¥ 1,100
Am9101APC	256×4bit static RAM500ns	¥ 1,000
Am9102APC	1024×1bit static RAM500ns	¥ 630
Am9102BPC	1024×1bit Static RAM 400ns	¥ 650
Am9111BPC	256×4bit Static RAM 400ns	¥ 1,100
Am9112BPC	256×4bit Statit RAM 400ns	¥ 1,100
Am1702ADC	256×8bit EPROM	¥ 3,200
Am2708	1K×8bit EPROM	¥ 14,000
P8212	8bit I/O Port	¥ 1,100
P8216	Quad Non-Inverting Bus Driver	¥ 900
P8226	Quad Inverting Bus Driver	¥ 900
P8228/P8223	System Controller	¥ 2,600
Am8224	CLOCK Generator and Driver	¥ 2,400
Am9551DC	Programmable Communication Interface	¥ 4,200
Am9555DC	Programmable Peripheral Interface	¥ 4,200
Am3341	64×4 FIFO	¥ 2,200
XTAL	18.432MHz	¥ 1,000

★NEWS／遂に500円を割りました。1K RAM P2102 …… ¥ 490

●amd DATA BOOK

MOS/LSI Data Book	¥ 2,500	〒300
Schottky And Low-Power Schottky	¥ 3,000	〒400
Linear And Interface Data Book	¥ 3,000	〒400

■御注文は下記アドバンスト・エクイップメント・リサーチへお申込み下さい。
■IC 送料は個数にかかわらず御注文1回につき一律200円加算して下さい。

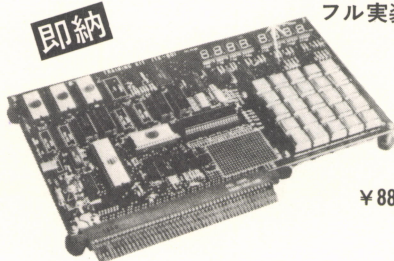
■マイクロコンピュータ専用電源 POWERFUL-80

¥15,000 〒1,000

■マイクロコンピュータ用に特別に設計されたコンパクトで高性能な電源です.
■＋5V，＋12Vの2電源が組込まれています。(5V 0.9A, 12V 0.15A)
■MK－80A，TK－80どちらにも使えます。
■パネルはブラック，ケースは黄色の美しい外装です。
■外形寸法：100×171×55(mm)

■MK-80Aキットの部品分売も致しております

プリント基板(大)	¥17,000	〒500
配線用プリント基板(小)	¥ 1,800	〒200
ユニバーサルプリント基板(小)	¥ 2,800	〒200
キー取付用アルミ板	¥ 2,000	〒200
KEYスイッチ	1ヶ@¥ 250	〒100
〃	10ヶ@¥ 220	〒200
キー用文字シール	一式 ¥ 500	〒 50

■NEC μCOMトレーニングキット TK-80のお取り扱いも致します。

CMOS RAM μPD5101CE
フル実装サービス中！

¥88,500 〒1,000

※お求めの方には16,400円相当のRAM及びICソケット(μPD5101CE ¥3,900×4と22PICソケット¥200×4)をサービス中！

■NEC マイクロコンピュータデバイス

μPD8080A	¥4,800	μPD8224	¥2,600
〃 454D	¥4,540	〃 8228	¥3,600
〃 5101CE	¥3,900	〃 8255	¥4,800
〃 2101C	¥1,200	〃 757	¥3,500
〃 2102AL-C4	¥ 850	〃 758	¥3,000
〃 8216	¥1,800		

その他の品種についてはお問合せ下さい。送料は個数にかかわらず御注文1回につき一律200円加算して下さい。

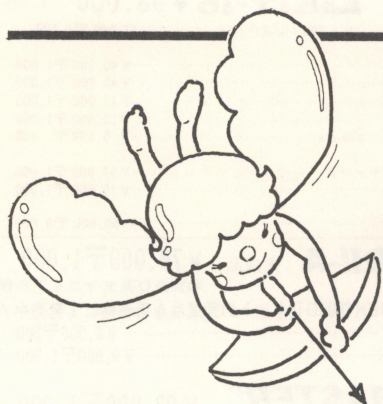
■TK-80のマニュアルのみの販売も致しております

■価格：TK-80概説 ¥90(100g)／TK-80ユーザズマニアル ¥590(500g)／μCOM-8・80プログラミング入門 ¥480(500g)／TK-応用プログラム ¥220(250g)／μCOMシリーズ総合ユーザス・ガイド ¥240(500g)／TK-80アプリケーションノート ¥710(500g)／プログラム・ライブラリNo.1～3 ¥90(250g)プログラム・ライブラリNo.4～11 ¥310(500g)／μCOM-8・80インストラクション活用表 ¥50(100g)／マイクロコンピュータ入門講座テキスト ¥700(500g)
■送料：100g～250g→〒200／250g～500g→〒650／500g～1kg→〒950／1kg～2kg→〒1,120／2kg～3kg→〒1,380／3kg～4kg〒1,400

■TK-80の修理承っております

お手持ちのTK-80動作トラブルでお困りの方は調整費1台につき¥20,000を添えて現品をお送り下されば2週間以内に完調のうえ御返送申上げます。破損部品があるときは部品代実費別途申受けます。

I/O係 ADVANCED EQUIPMENT RESEARCH CORP.
(アドバンスト・エクイップメント・リサーチ)

営業時間 AM10:00～PM6:00
定休日 毎週 月・木曜日

新しいスタイルのマイコンショップ

COMPUTER Lab. コンピュータ ラブ

あなたはマイクロコンピュータで何をやってみたいですか？どこのどんなチップを使いますか？参考になる本は，回路はリレーやモーターのドライブ，装置のコントロールは？ロボットは？プログラムのことは忘れていませんか？作り方は，アセンブラはベーシックは，そしてP－ROMの書き込みは？じっくり取り組みたい方の“のんびりショップ”です．いろいろ迷っている方も，ぜひおいで下さい．機種は実際に確かめてから，決めて下さい．

営業内容

1)マイクロコンピュータ，RAM，ROM及び関連品
●モステクノロジ・インターシル・MMI・インテル・富士通・NS

2)ローパワーRAMメモリ
Kit(500nS) 定電圧回路(2K毎)アドレス／データバッファ，番地セレクタ，バッテリバックアップ用回路付，S－100バス構成，全メモリソケット付，各社マイクロコン・キットに使用可能

価格		
最大	8K	¥85,000
最小	2K	¥53,800
メモリ	1K分	¥ 5,200

3)ソフトウェア TINY BASIC
6800用，6502用 価格 ¥3,000

4)P－ROM書込サービス
価格 1KビットROMコピー ¥1,000
TTY使用料1時間 ¥1,000

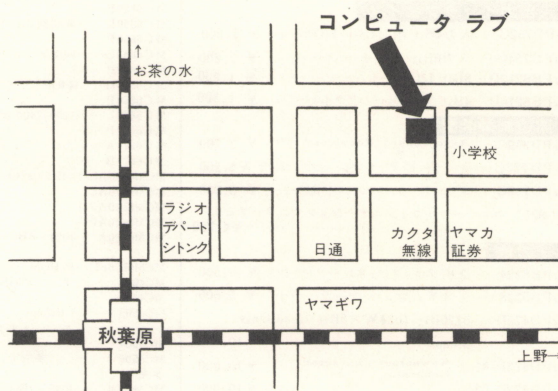
5)マイコンセミナ
(日) TINY BASIC 10：00～12：00(月3回) ¥6,000
(月) TINY BASIC 18：30～20：30(月3回) ¥6,000
(木) アセンブラ(6502) 18：30～20：30(月4回)
入会金 ¥3,000，月謝 ¥7,000

サウスウエスト・テクニカル・プロダクツ 代理店

人材募集

マイコンによる制御機の設計に興味のある方．
〔詳細は☎(03)816－3911〕

営業時間

◇火，水，金 13時～19時
◇土，日，祭日 11時～17時
◇セミナー 月，木，日

千代田特殊無線（株）
（株）イーエスディ ラボラトリ

東京都千代田区外神田3－3－4 〒101 千代田特殊無線ビル4F ☎(03)253－0737／816－3911

NEC μCOM Training Kit TK-80

¥88,500〒1,000

■オプション
- ●CRTディスプレー完成品……¥33,000〒1,500
- ●CRTディスプレー部一式（基板ユニバーサル）……¥15,000〒1,000
- ●カセットインターフェイス（部品一式）……¥ 900〒 140
- ●専用プリンター　放電プリンターEUY-10E　¥16,000〒1,000
- 〃　ドライバー　¥13,000〒1,000
- ●TTY　インターフェイス部品一式……¥ 750〒 150
- ●定電圧電源完成品……¥ 9,800〒1,000
- ●TK-80　アプリケーションノート……¥ 710〒 350

16Bit Microcomputer Kit パナファコム

LKIT-16（エルキット16）¥98,000〒1,000

■オプション　MB8111……¥1,000　定電圧電源……¥17,000〒1,500

4K-RAMボード
- ボード完全キット……¥40,000〒1,000
- ボード完成品……¥45,000〒1,000
- ボードのみ……¥12,000〒1,000
- RAMなしボードキット……¥18,000〒1,000
- 1K-RAM（2102×8）……450n/s　¥ 5,500〒 200

TV-CRTデスプレー
- CRTデスプレー（2513キャラクター使用）完全キット……¥37,000〒1,000
- CRTボードのみ……¥15,000〒1,000

プリンター
- PR-40（コントローラ付）……¥150,000〒3,000

HITACHI トレーニングモジュール

■オプション　HM472114……¥4,200

H68/TR ¥99,500〒1,000

《特長》●ファームウェアによる本格的アセンブラを内蔵　●14桁蛍光表示管で見易いミドリ色（変形アルファベット表示）　●48英字記号のフルキーボード　●テキストエディタ使用可能　●オーディオカセットテープレコーダー（300ボー）の自動START/STOPが可能　●カンサスシティスタンダード　●ベーシックIKB大容量メモリ（オプションで2KB追加できます）　●0.8msハードウェアタイマ割込み回路付　●バスの信号配列を標準化しオプション用ボードとの接続が容易　●5V単一電源　●電源、テープレコーダー専用コネクタ装備

MEK6800DII-A（完成品）¥79,000〒1,000

和文及び英文マニュアル付

無限の拡張性を秘めたMEK6800DIIキットの完成品を新価格にて発売中！
- オプション専用コネクター……¥2,500〒500
- 専用定電圧電源……¥9,900〒1,000

"SPEED MASTER" MEK6800DII-B ¥93,000〒1,000

和文及び英文マニュアル付

貴方はパワースイッチをONするだけ。M6800システムをスピーディーにマスター出来ます。

プリンターコントローラ用
- MK2302P…キャラクタ・ジェネレータ、5×7DOT 64文字　PRINTER用……¥5,500
- F3257ADC…キャラクタ・ジェネレータ、5×7DOT 64文字　PRINTER用……¥7,000
- F3351……FIFO、40×9Bit……¥3,800
- TMS4024…FIFO、64×9Bit……¥6,000

モステック富士通
- Z-80（MK3880N）CPUマニアル付　¥12,000〒250
- MK3881N　¥ 5,500〒250
- MK3882N　¥ 5,500〒250
- MN1630N　¥ 6,500〒250
- MB8101　¥ 950〒250
- MB8111　¥ 1,000〒250
- MB8518　¥15,000〒250
- 2513（G・I社）単一電源　¥ 4,800〒250

CPU
- μPD751D……（μCOM-4）4-Bit CPU……¥ 7,500
- μPD8080A…（μCOM-8）8-Bit CPU……¥ 5,000

ROM
- μPD454D……256W×8　P-ROM……¥ 7,000

RAM
- μPD412C……256W×4スタティックRAM…¥ 2,000
- μPD2101C……256W×4　〃……¥ 1,200
- μPD2102AL-4　1024W×1Bit　フルデコード 1024Bitスタティック RAM450n/s　¥ 950
- F2102-(1)……1024Bitスタティック RAM350n/s　¥1,300
- μPD5101CE…256W×フルデコード1024Bit スタティックRAM……¥ 3,500
- 1101……（マイクロシステム社）……¥ 350
- 2102……450n/s……特価8本組　¥ 5,500

入出力インタフェイス
- μPD752C……入力4Bit 出力4Bit I/Oポート…¥ 1,200
- μPD754C……入力8Bit ラッチ……¥ 2,200
- μPB8212D…8Bit I/Oポート……¥ 1,200
- μPB8216D…4Bit 双方向バスドライバ……¥ 1,500

周辺制御装置
- μPD369C……Asyncronors Receiver/トランスミッタ　¥ 3,700
- μPD757C……キーボード・ディスプレーコントローラ　¥5,200
- μPD758C……プリンターコントローラPRC……¥ 3,300
- P-8251　ユニバーサルコミュニケーションインターフェイス……¥4,310

その他
- μPB8224……2相クロックジェネレータドライバ　¥ 3,600
- μPB8228……システムコントローラ　¥ 5,600
- μPD472D……5120Bit（1024W×5Bit）Read Only Memory……¥10,000
- μPD473D-01…Rowontput Character Generator　¥10,000
- μPD473D-02…　〃……¥10,000
- μPD474D-01…Column Output Character Generator…¥10,000
- μPD474D-02…　〃……¥10,000
- μPD8255……¥ 6,000
- モステック　MK4096…4096×1Bit　ダイナミックRAM……¥ 1,200

- μPB8214……8080A用インタラプタコントローラ¥ 4,500

シグネティックス社　キャラクタージェネレーター
- 2513……英文字　64文字……¥4,800
- カナ文字　64文字……¥4,800

キーボード
- KBR-014……¥55,000〒2,000　フルキーボード・キー数：63キー（MAX72キー）英数、カナ、モード外部制御可能　JIS-6220
- KBR-015……¥61,500〒2,000　テンキー付フルキーボード・キー数：74キー（MAX91キー）・英数、カナ、モード外部制御可能　JIS-6220　8単位符号　大数パリティー
- プリンター　DMTP-6　OEM……¥ 200,000〒2,000（モジュールはインパクト方式のドットマトリックスによるシリアルプリンタ）

つくるコンピュータ　μCOM-4
- μPD751D×1……28Pソケット付
- μPD2102×4……16P　〃
- C-MOS4001×1…14P　〃
- C-MOS4011×1…14P　〃
- C-MOS4016×2…14P　〃
- C-MOS4028×1…16P　〃
- C-MOS4042×1…16P　〃
- C-MOS4050×1…16P　〃
- C-MOS4076×1…16P　〃
- 発光Di×6……ホルダー付
- μA78L12×1
- IS1588×2
- μA7805×1
- IN60×2
- ノイズフィルター×1
- 10D-1×4
- ユニバーサル基板×1（NEC・紙エポキシ 22Wコネクター付）

上記μCOM-4部品一式を¥21,000〒500

モトローラ
- MC6800L……8Bitパラレル処理プロセッサ……¥ 8,600
- MC6800P……〃……¥ 7,250
- MC6802……MICROPROCESSOR WITH CLOCK AND RAM……¥15,000　6800+6871+6810
- MC6820L……16Bit（8Bit×2）パラレルインタフェイス（PIA）…¥ 4,250
- MC6820P……〃……¥ 3,250
- MC6850L……非同期式シリアルインタフェイス（ACIA）¥ 4,250
- MC6850P……〃……¥ 3,250
- MC6852L……同期式シリアルインタフェイス（SSDA）¥ 6,120
- MC6852P……〃……¥ 5,500
- MC6860L……Q-600ボーモデム……¥ 6,120
- MC6860P……〃……¥ 5,500
- MC6862L……1200/2400ボー・DPSK変調器……¥ 6,120
- MC6862P……〃……¥ 5,620
- MC6871A……クロックジェネレータ……¥ 8,100
- MC6871B……〃……¥ 7,200
- MCM6810AL-1　128×8Bit　RAM……¥ 3,250
- MCM6810AL……〃……¥ 2,500
- MCM6810AP-1……〃……¥ 2,350
- MCM6810AP……〃……¥ 1,800
- MCM6830A…1024×8Bit・マスクROM……¥ 5,000
- MCM68708L……P-ROM……¥18,000
- MC4044……クロック同期用PLLキット……¥ 1,100
- MC4024……〃……¥ 1,100
- MC8503……CRCチェッカー／ディテクタ用……¥ 4,300
- ユニバーサル多項式ジェネレータ（4Bit）
- MC8504……〃（8Bit）¥ 3,300
- MC8506……〃　多項式ジェネレータ（16Bit）……¥ 6,200

メモリ
- MCM6604……4096×1Bit 16ピン　RAM……¥ 2,500
- MCM6605A……4096×1Bit 22ピン　RAM……¥ 3,500

キャラクタージェネレータ
- MC6573AP……¥ 4,800

インタフェイス用　LSI
- MPQ6842……MPU クロックバッファ……¥ 1,600
- MC8T26……バスドライバ……¥ 1,200
- MC8T96P……〃……¥ 850

- MC1488……ラインドライバ……¥ 1,400
- MC1489……ラインレシーバ……¥ 1,400
- MC3459……メモリ・アドレス・ドライバ……¥ 1,500
- MC3460……メモリ・クロック・ドライバ……¥ 1,700
- 8T95……¥ 850
- DM8097（8T97フルコンパチ）……¥ 350
- 8T98（8T98　〃　）……¥ 350

モトローラ技術資料
- M-6800……MPU Application Manual……¥6,000〒500
- M-6800……MPU Programming Manual……¥1,500〒300
- M-6800……マイクロコンピュータマニュアル……¥1,800〒300
- C-MOS……データ Book……¥1,000〒300
- リニヤIC　データBook……¥1,500〒300

東芝マイクロコンピュータ
- TLCS-12A　EX-O……¥99,000〒1,300
- ワンボードマイクロコンピュータ TLCS-12A　EX-12/10……¥185,000〒1,000
- TLCS-12Aコントロールパネル（オプション）……¥178,000〒1,000
- SDK-80……¥83,000〒1,000
- LKIT-8……¥85,000〒1,000

テキサス
- SN74S188N……32×8　P-ROM……¥ 1,000
- SN74S287N……256×4　〃……¥ 1,500
- SN74S387N……256×4　〃……¥ 1,200
- SN74S470N……1024×8　マスクROM……¥ 2,200
- SN74S472N……512×8　P-ROM……¥ 6,000
- TMS2708JL……MOS 8K　EP-ROM……¥12,000
- B1702-6……2048Bit　P-ROM……¥ 3,950
- TMS4035NL……1024×1　スタテック RAM……¥ 1,200
- TMS4036NL……64×8　スタテックRAM……¥ 3,200
- TMS4039NL……256×4　スタテックRAM……¥ 2,100
- TMS4042NL……256×4　スタテックRAM……¥ 2,100
- TMS4043NL……256×4　スタテックRAM……¥ 2,100
- TMS4050NL……4096×1　4Kダイナミック RAM……¥ 2,400
- TMS4060NL……4K ダイナミックRAM……¥ 2,500
- TMS4044NL-45…4K スタテックRAM……¥ 6,500
- TMS4045NL-45…4K スタテックRAM……¥ 6,500
- TMS4046NL-45……¥ 6,500
- TMS4047NL-45……¥ 6,500
- SN74LS240……¥650　SN74LS244……¥650
- SN74LS241……¥650　SN74S 240……¥1,090
- SN74LS242……¥630　SN74S 241……¥1,090
- SN74LS243……¥630

インテル
- P8080A……8Bit Central Processor Uuit（2μs Cycle）……¥ 6,420
- P1702A-6 Unprogrammed　Hermetic Erasable and Electrically Reprogrammable 2048 Bit PROM（1.0μs）　¥ 3,950
- P2115……High-Speed Static 1024 Bit/Open-Collector RAM　¥ 5,620

日立
- HM435101-1G……（256W×フルデコード 1024Bit スタテックRAM）……¥ 3,000
- HM435101-1P……¥ 2,320
- HM4704-2……RAM……¥ 2,350
- HM4711-3……RAM……¥ 2,700
- HN351702……P（ROM）……¥ 6,900
- MD46505……CRTコントローラ……¥15,800

DIP　ソケット			
8P……¥ 80		22P……¥180	
14P……¥ 70		24P……¥200	
16P……¥ 80		28P……¥250	
18P……¥ 120		40P……¥300	
20P……¥ 150		42P……¥350	

DIPラッピング用ソケット	
14P	¥ 220
16P	¥ 240
24P	¥ 470
28P	¥ 550
40P	¥ 740
42P	¥ 810

秋葉原駅前ラジオ会館4F

株式会社 若松通商 I/O係

- 秋葉店店　〒101　東京都千代田区外神田1—15—16 秋葉原ラジオ会館4階　☎03(255)5064
- 通販部　〒211　神奈川県川崎市中原区小杉陣屋町1-547-80　☎044(722)0948

※指定以外の送料200円

各社IC・半導体・パーツ

求む　営業2名　経理1名（詳細面談）

NEC μCOM Training Kit TK-80

定価￥88,500〒1,000
（メーカー完全保証キット）

■主な仕様
CPU……… 8080A（2,048MHzで動作）
PROM…… 454D 0.75kバイト（最大1Kバイトボード上）
RAM……… 5101C 0.5Kバイト（最大1Kバイトボード上）
信号………. データバス(8) TTL入力／出力
アドレスバス(16) MOS出力（レベルはTTL）
I/O端子(24) 8ビット×3ポート
電源……… ＋5V、＋12V
消費電流… 0.9A以下、0.15A以下
■拡張用
SN7400…………￥70
IS953……………￥20
μPD4049…… ￥300
μPD454D …￥5,000
μPD2101 …￥1,100
μPD2102 …￥　800
μPD5101 …￥4,000

MEK6800DII（モトローラ8Bit）…￥79,000
MCM6810￥1,800　MC8T26 ￥ 1,200
MC8T96 ￥　900
LKIT-8（富士通8Bit）……………￥85,000
MB7054…￥　MB8112‥￥　900
LKIT-16（パナファコム16Bit）……￥98,000
MN1630…￥6,500　MB8111‥￥ 1,100
MB8518‥￥ 9,000
EX-5（東芝12Bit）………………￥77,000
TMM121C ￥6,000　T3219……￥ 6,500
50Pコネクタ￥
HMCS-6800…………………￥99,500
（日立8bit 5V単一）
CRTディスプレー（NEC）………￥33,000
共通オプション
EUY-10E（放電プリンター）……￥16,000
EUY-PUD（プリンタードライバー）‥￥13,000
キャラクタージェネレーター
2513…CM2770（英文）…………￥ 4,500
CM4800（カナ）
TMS6011（UART）……………￥ 2,500

低価格ツールの登場！
より確かな精度を実現

■WRAPET
ebr 800/WTI ￥9,800〒300
■ラペット専用ビット
ebr830/ebr28(0.32)
ebr30(0.26)
￥2,200〒100
■スタンダードビットスリーブ
（各社共通用）
ebr830/SB ebr30(0.26)￥4.800〒140
828/SB 〃 28(0.32)￥4.800 〃
826/SB 〃 26(0.4)￥4.800 〃
824/SB 〃 24(0.51)￥4.800 〃
■アンラップツール
ebr810(ebr30、28用)￥2,200〒140
ebr810(ebr26、24用)￥2,200〒140
(0.26φ、0.32φのコードも在庫あり)
カタログ〒100

高感度保護回路内蔵　安定化電源　定価￥7,000〒500

● DC13.8V（7.5V～14V可変可能）● DC3.2A（DC4Aにて保護回路作動）
● 寸法：120W×80H×160D㎜　● アマチュア無線、CB、実験用に最適

新登場！ M51845L50時間タイマーIC￥1,000 説明書〒200

《特長》
1）最大50時間
2）時限設定は可変抵抗1本にて可能
3）安定化用ツェナーDi内蔵
4）TTL ICと接続可能
5）電源ON時のリセット回路内蔵
6）小型、低消費電力
Vcc：6V、1Z：15mA
Pd：360mW、Topg：0～60℃

TDKスイッチングパワーサプライ　￥49,500〒500

	DC出力			AC出力
TRM001	＋5V 10A	＋12V 1A	－12V 1A	6.3V 0.1A
TRM002	〃	＋15V 1A	－15V 1A	〃
TRM003	〃	＋12V 1A	－ 5V 1A	〃
TRM004	〃	＋15V 1A	－ 5V 1A	〃
TRM005	〃	＋12V 1A	－ 9V 1A	〃

	DC出力		
TRM021	＋5V 5A	＋12V 0.3A	－12V 0.3A
TRM022	〃	＋15V 0.3A	－15V 0.3A
TRM023	〃	＋12V 0.3A	－ 5V 0.3A
TRM024	〃	＋15V 0.3A	－ 5V 0.3A
TRM025	〃	＋12V 0.3A	－ 9V 0.3A

NEMIC（日本電子メモリ工学）社スイッチングレギュレーター取扱中　資料〒200

サイエンス・システム・サポート社製品取扱中

山一ICソケット

	半田付用	価格	ラッピング用	価格
8P	IC66－22	140		
	IC109－22	100	IC107－14＃1W	200
14P	IC09－22	110	IC09－14＃1W	220
	IC09－24	120		
16P	IC31－22	140	IC31－14＃1W	240
	IC31－24	150		
18P	IC25－2203	150	, C25－1403＃1W	360
	IC25－2403	170		
20P	IC40－20	200		
22P	IC102－22	210		
	IC102－24	190		
24P	IC49－2206	200	IC49－1406＃1W	470
	IC49－2406	220		
28P	IC99－22	220	IC99－14＃1W	530
	IC99－24	250		
36P	IC70－22	270		
40P	IC63－22	300	IC63－14＃1W	740
	IC63－24	320		
42P	IC100－22	320	IC100－14＃1W	850
	IC100－24	350		
	TO5用ソケット		デイクリート	
TO5－8P	IC03－3A	170	IC09－91	330
－10P	IC02－4A	180	IC31－91	360
－12P	IC16－2A	190		
	LED用		シングル8P	140
313用	LED1012－9＃2W	160		
306用	LED1013－9＃2W	200		

A/Dコンバータ
8700CJ（8bit）……………￥ 5,600
8701CN（10bit）……………￥13,800
8702CN（12bit）……………￥18,500
D/Aコンバータ
μpc603D（6bit）……………￥ 3,800
μpc610D（8bit）……………￥
3½ A/Dコンバータ
8750CJ……………………￥ 5,500
8750CN……………………￥ 8,200
MC14433 …………………￥ 3,500
V/F-F/Vコンバータ
9400CJ（Dip14p）…………￥ 3,600

新発売！ TPS-7706A…￥37,000

MINI DOT PRINTER UNIT 〒500

●EUY10E使用
●5×7ドットマトリックス方式
●32桁・40桁
（½桁切換スイッチ付）
●64キャラクター
●LKIT-8、MEK6800DII
TK80との接続図付
●ロール紙　￥500

汎用 OP-Amp
LM741CH ￥180　RC4558D（Dip-8P）￥270
LM301AH ￥130　RC4558T（TO5-8P）￥300

単一電源用 OP-Amp

	電源電圧	動作温度	価格
μPC324C	3～30V	0～＋70℃	550
CA324	3～30V	－55～＋125℃	700
LM324N	3～30V	0～＋70℃	1.200
LM2900N	4～30V	－40～＋85℃	550
LM2902N	3～$26V_{max}$	－40～＋85℃	600
LM3900N	4～30V	0～＋70℃	450
CA3130T	5～15V	－55～＋125℃	450
CA3130AT	5～15V	－55～＋125℃	1,800
CA3140T	4～30V	－55～＋125℃	500

（3130、3140以外はQuad OP-Ampです。
2900、3900は電源差動入力 OP-Ampです。）

マイクロパワー OP-Amp

	電源電圧	動作電圧	価格
CA3078T	±1.3～±6V	0～＋70℃	900
μA776HC	±1.2～±15V	0～＋70℃	1,400
μPC153A	±3 ～±15V	－20～＋70℃	1,390
LM4250CH	±1 ～±15V	0～＋70℃	

FET入力 OP-Amp
LF356CH……高速高帯域OP-Amp…………￥700
LF357CH……　〃　…………￥700
μPC152A……超高入力インピーダンス…￥1,800
TA7505M……　〃　￥1,900

モトローラC・MOS

品番	価格	品番	価格	品番	価格	品番	価格	品番	価格
				MC14070	￥ 100	MC14508	￥1,000	MC14539	￥ 600
				MC14071	￥ 100	MC14510	￥ 670	MC14541	￥ 630
				MC14072	￥ 100	MC14511	￥ 750	MC14543	￥ 770
				MC14073	￥ 100	MC14512	￥ 540	MC14549	￥2,160
				MC14075	￥ 100	MC14514	￥1,500	MC14552	￥6,500
MC14000	￥ 100	MC14027	￥ 250	MC14076	￥ 570	MC14515	￥1,500	MC14553	￥2,250
MC14001	￥ 100	MC14028	￥ 470	MC14077	￥ 100	MC14516	￥ 620	MC14554	￥ 760
MC14002	￥ 100	MC14032	￥ 610	MC14078	￥ 100	MC14517	￥2,970	MC14555	￥ 420
MC14006	￥ 600	MC14034	￥1,900	MC14081	￥ 100	MC14518	￥ 650	MC14556	￥ 420
MC14007	￥ 100	MC14035	￥ 600	MC14082	￥ 100	MC14519	￥ 380	MC14557	￥2,300
MC14008	￥ 550	MC14038	￥ 680	MC14160	￥ 630	MC14520	￥ 680	MC14558	￥ 670
MC14011	￥ 100	MC14040	￥ 620	MC14161	￥ 630	MC14521	￥1,250	MC14559	￥2,160
MC14012	￥ 100	MC14042	￥ 420	MC14162	￥ 630	MC14522	￥ 670	MC14560	￥ 790
MC14013	￥ 200	MC14043	￥ 400	MC14163	￥ 630	MC14526	￥ 670	MC14561	￥ 340
MC14014	￥ 600	MC14044	￥ 390	MC14174	￥ 590	MC14527	￥ 670	MC14562	￥2,970
MC14015	￥ 600	MC14046	￥ 650	MC14175	￥ 540	MC14528	￥ 430	MC14566	￥ 840
MC14016	￥ 200	MC14049	￥ 190	MC14194	￥ 640	MC14529	￥ 800	MC14568	￥ 0
MC14017	￥ 600	MC14050	￥ 190	MC14501	￥ 100	MC14530	￥ 460	MC14572	￥ 190
MC14020	￥ 640	MC14051	￥ 500	MC14502	￥ 410	MC14531	￥ 840	MC14580	￥3,400
MC14021	￥ 600	MC14052	￥ 500	MC14503	￥ 350	MC14532	￥ 660	MC14581	￥1,750
MC14022	￥ 550	MC14053	￥ 500	MC14505	￥2,750	MC14534	￥3,500	MC14582	￥ 730
MC14023	￥ 100	MC14066	￥ 260	MC14506	￥ 170	MC14536	￥1,500	MC14583	￥ 490
MC14024	￥ 420	MC14068	￥ 100	MC14507	￥ 180	MC14537	￥8,400	MC14585	￥ 760
MC14025	￥ 100	MC14069	￥ 100						

※送料5,000未満〒200、5,000円以上〒350と定め、多い場合は返金致します。

I/O係

本社直販部・通販部　AM10：00～PM6：00振替口座　東京3-70584
〒145 東京都大田区田園調布本町57－1　私書箱3号　☎03（722）0844
神田営業所　AM10：30～PM6：00　（交通博物館となり、ラジオガーデン内）
〒101 東京都千代田区神田須田町1－25　☎03（251）1201

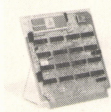
Let me introduce my

Chibicom

チビコン

チビコンシリーズは汎用マイクロコンピュータとして高信頼設計のもとに要求に応じたシステム構成をとることのできるコンポーネントスタイルです。

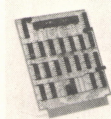
NO.1(a)マザーボード
2.54㎜ピッチ72Pコネクタを使用
コモンバス方式のマザーボード
10本までのコネクタを実装可能
ラインコネクタを接続することにより増設をボード枚数単位で可能
5カードコネクタ実装　完成品¥14,000

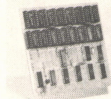
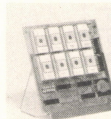
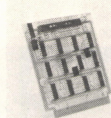
NO.1(b)カードラック
上記マザーボード用標準カードラック
小形ユニバーサルカードラックを使用。
5カードコネクタ標準ラック付　完成品¥18,000

NO.2 CPU-1 6800用
標準品はCR発振1MHzまでのクロックを使用
オプション—クリスタル発振8MHz 1/8分周データバスドライバーとし8T26(ファンアウト30)アドレスバスドライバーとして74366(ファンアウト20)を使用　CPUチップ実装　キット¥19,800

NO.3 CPU-2 8080A用
標準品はCR発振約6MHzのクロックを使用。
オプション—クリスタル発振8MHz1/4分周。(バスドライバーは6800No.2ボードに同じ)
CPUチップ実装　キット¥19,800

NO.5 SC/MP
8080A用No.3ボード仕様と同じ
CPUチップ実装　キット¥15,800

◀NO.22 CRT-1F
家庭用のテレビをディスプレイとして利用
7×9ドット、英、数、カナ記号の全てが表示。グラフィックにも使用可能
カーソル機能2ページ分のメモリー、よこ32文字たて16行、512文字を1ページに表示、RFによるアンテナ端子入力でありテレビを無改造にて使用(ライトペンも可能)
ケース電源無し　キット¥42,000

▼NO.25 αプリンタ(放電及び感熱)
放電及び感熱プリントメカを使用
5×7ドットにて最大40桁の英数記号を記録。AC電源内蔵、8ビットのデータ入力により印字可能
TK—80等メーカーキットにインターフェス無しで接続可能
放電プリンタ仕様　キット¥59,800
感熱プリンタ仕様　キット¥63,800

NO.6 KIT-1F TK-80用
日電TK—80トレーニングキットのインターフェスカードを利用することによりROM.RAM.I/O等の拡張がチビコンシリーズで可能

NO.9 ユニバーサルボード
ハンダディップ用のユニバーサルカードでありI/O等の特別な仕様を組むことが可能。

NO.12 RAM-2 2102用
1K×1bitRAMである2102を使用
2KBのメモリーを構成
メモリーアドレスは2K単位でジャンパーにて自由に設定。
2102AL-4 2KB実装　キット¥23,800

NO.14 ROM-1 1702A用
256×8bit ROM 1702Aを使用2KBのメモリーを構成。
1702Aコンパチブルのマスク ROM使用可能
1702A256B/T実装　キット¥11,200

NO.16 I/O-1 パラレルI/O
8ビットパラレルの入出力がそれぞれ2TTLレベルにて入出力(ラッチ出力)I/Oアドレスはジャンパーにて自由に設定

取扱い代理店／日本インターナショナル整流器(株)・サンエイ(株)他.　■取扱い販売店募集中
お詫び：チビコン総合マニュアル御希望の方多数のため発送が遅れております。あとしばらくお待ち下さい。
お問い合せ：〒532 大阪市淀川区西宮原2丁目6-16-201 Phone：06-395-1571 データアドバンストプロダクツ株式会社
お知らせ：9月より右記住所に移転いたします。〒530 大阪市北区玉江町2-2 大阪国際貿易センター内

DATA PRO
DATA ADVANCED PRODUCTS INC.

ロジックシステムズ　インターナショナル社製

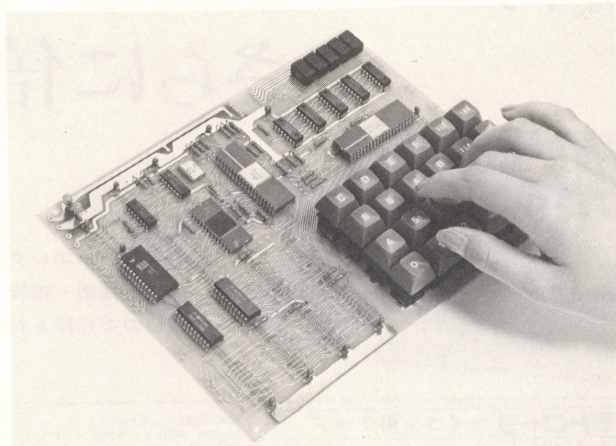
MP-80

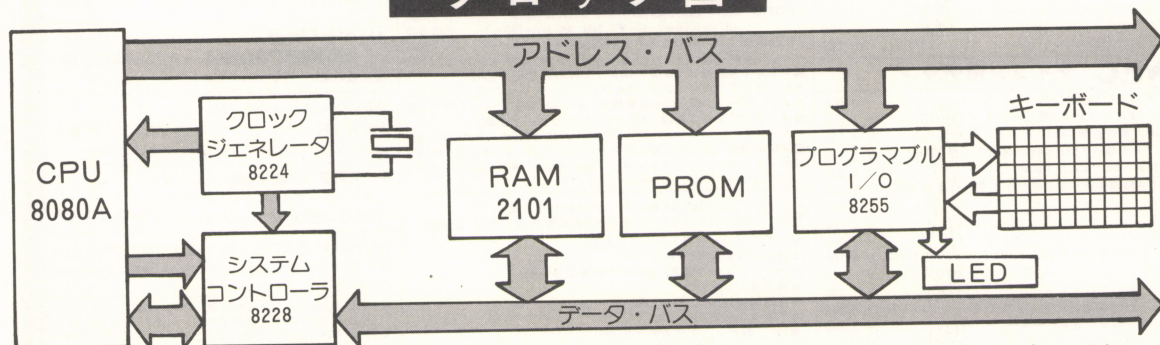
トレーニング・キット
(マニュアル付)

¥39,500!(〒500)

驚異のコスト・パフォーマンスを実現しました。

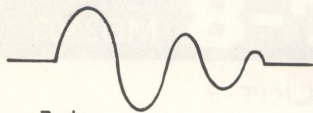
(MP-80 vs TK-80比較表)

	MP−80	TK−80
CPU	8080A	8080A
クロック	2.048MHz	2.048MHz
RAM	最大 1KB 実装 256B	最大 1KB 実装 512B
P−ROM	最大 512B (モニタ 256B付)	最大 1KB (モニタ 768B付)
表示器	7セグメント LED	7セグメント LED

☆**拡張用RAM　256B分**(2101,2個)
特価¥2,000(〒200)
☆**専用電源　¥13,000**(〒500)

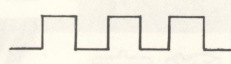
ブロック図

MP-80マイコン・キットは通信講座「作りながら覚えるマイコン入門コース」のために用意された組立てキットです。通信講座については主催のマイテック社／TEL 03(661)3366へお問合せ下さい。

外付拡張用4KバイトRAMボード(1KBのみ実装，動作テスト済)
(マニアル付)

●アクセス・タイム　500ns
●寸　法　130mm×165mm
●コネクタ　2.54mmピッチ，両面88極¥27,500(〒500)
☆拡張用RAM　1KB分(2102，8個)　特価¥5,200(〒300)

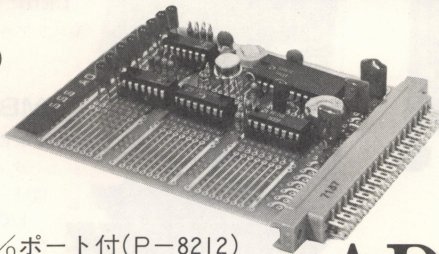
杉山商事(有) I/O係　〒113 東京都文京区千駄木3-52-9(田島ビル)　TEL 03-821-3245

Low Cost!

入力

アナログ・データ
&スタート・パルス

出力

8 Bitバイナリー
&終了フラグ

DATA ACQUISITION
FOR : 8080
6800

・入力 0～＋10V
（変更可能）
・変換速度 5 μs
・直線性 ½LSB
・クロック内蔵
・シリアル出力付
・トライステート1/0ポート付（P－8212）
・デジタル出力モニタ（LED 8 個）取付可能

広いフリー・エリアには、サンプル・ホールド，アナログ・スイッチ，ＯＰアンプ等の追加装備が可能です。

μAD-08kit ¥15,500円

詳細マニュアル付（〒サービス）

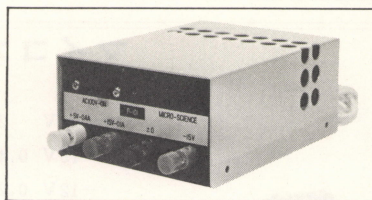
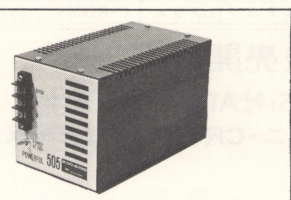
DAコンバータ（バイナリー入力） 〒100

	品名	出力	セットリング・タイム	パッケージ	メーカー	価格
8 Bit	AmDAC-08CQ	電流出力	150ns	16ピンDIP	AMD	¥ 3,510
8 Bit	AmDAC-08EQ	電流出力	150ns	16ピンDIP	AMD	¥ 4,300
12Bit	DAC80-CBI-I	電流出力	300ns	24ピンDIP	マイクロ・ネットワーク	¥ 9,900
12Bit	DAC80-CBI-V	電圧出力	3 μs	24ピンDIP	マイクロ・ネットワーク	¥10,950

逐次比較型ＡＤコンバータ用ＩＣ		〒100
LM311H	コンパレータ	¥ 700
Am2502PC	8 Bit逐次比較用レジスタ	¥ 4,500
Am2504DC	12Bit逐次比較用レジスタ	¥ 5,500
Am8212PC	8 Bit I/Oポート（トライステート）	¥ 1,200

マイコン、計測用安定化電源Powerfulシリーズ標準仕様
リップル0.5mVrms以下、レギュレーション0.5%以下。

4KバイトRAMボード
・1KBのみ実装（動作テスト済）
・アクセス・タイム 500ns
・寸法130W×165L
・コネクタ2,54mmピッチ両面88極
¥27,500（〒500）

★拡張用RAM2102型（500ns）
1KB分（8 個）¥5,200（〒300）

Powerful－101K（キット）
・＋5 V，400mA
・＋15V，100mA
・－15V，100mA
¥9,500（〒500）

Powerful－505
・＋5 V，3 A
¥18,000（〒サービス）

Powerful－515
・＋5 V，3 A
・－5 V，100mA
・±15V，200mA（9～16V可変）
¥24,500（〒500）

★各カタログ¥100円。送料（部数に関係なく）〒100円、切手可。
★大阪近郊の方は東亜無線電機（浪速区日本橋筋 5－61）で御覧下さい。
★地方代理店募集中 東京近郊の方はダイデン商事店頭で御覧下さい。

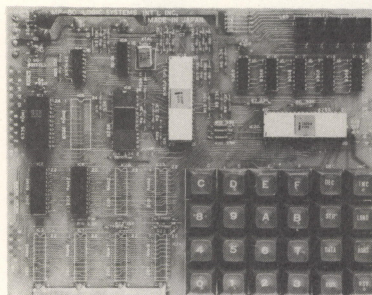
SSS ㈱サイエンス・システム・サポート
〒160 東京都新宿区新宿4－3－1
和宏ビル404号
TEL 03（354）1465

未来技術を先どりし、自己開発に役立つ

監修　東京大学教授　渡辺　茂
編集　電子技術総合研究所　矢田光治

マイクロコンピュータ通信教育

入門から製作と応用までがマスターできる
ユニークでシステマテックな８ビット中心の３コース

マイクロコンピュータへの入門……いろいろな入門があります。

基本教材 MP-80 マイコンキット
￥39,500（希望者別売）

作りながら覚えるマイコン　全７講３ケ月入門コース

ひとつの洗練された8080Aのキットをじっくり自作しながらマイクロコンピュータに関するいろいろなエッセンスを習得する入門。つまり１を知って10を知るという入門。

講座内容

Ⅰ．マイコンを作るには—エレクトロニクス入門
マイコンとは／部品の知識と集め方／工具とその使い方

Ⅱ．マイコンを作ってみよう
部品を取付ける／配線をする

Ⅲ．マイコンを働かしてみよう
マイコンの操作／マイコンを働かしてみる

Ⅳ．プログラムを組んで働かしてみよう
プログラム／フローチャートの作り方とプログラム

Ⅴ．マイコンのしくみはどのようになっているか
情報のあらわし方と取扱い方／エレクトロニクスの回路とマイコンの構造／プログラムはどのように実行されるか

Ⅵ．マイコンを使って音楽の自動演奏を楽しんでみよう
電子楽器の自動演奏の実験／モニタプログラムの入った部品の取付け／テープレコーダにプログラムを入れる

Ⅶ．さらに進んで勉強したい人のために
ソフトウェアのテクニック／進んだプログラムの体系／今後の勉強の進め方

受講料　￥26,000
執筆者
ロジック・システムズ・インターナショナル
石田　芳

基本教材 MP-68 マイコンキット
￥198,000（希望者別売）

使いながら覚えるマイコン　全８講４ケ月入門コース

今さらハンダづけやらマシン語でもあるまい。興味はもっぱらマイコンの高級プログラミングと自分の日常活動へのアプリケーションにあるというアプローチ。つまりマイコンを使い切ってマイコンの本質を理解する入門。

講座内容

Ⅰ．マイコンとは
マイコンの用途／構成と部品・機器／ソフトとプログラムの働き

Ⅱ．マイコンの作り方と働かし方
製作のポイント／組立てと配線／組立て後のチェック／マイコンの実行

Ⅲ．プログラミング言語BASIC
BASICとは／簡単なプログラムの例／サブルーチンとその例

Ⅳ．BASICの応用例
万年カレンダーの例／電子サイコロの例／アメリカのマイコン野郎が作ったBASICゲームの例

Ⅴ．マイコンのしくみ
レジスタの機能／命令とその機能／ブート・ストラップとバイナリ・ローダの働き／モニタプログラムの働き

Ⅳ．周辺装置とインタフェース
周辺装置の使い方／インタフェースの使い方と拡張

Ⅶ．アセンブラ
アセンブラとは／アセンブルのしかた

Ⅷ．電子楽器と人工頭脳への挑戦

受講料　￥28,000
執筆者
サウス・ウエスト・テクニカル・プロダクツ・ジャパン
新津政博

マイクロコンピュータ製作と応用　全15講５ケ月初級コース

マイコン作りの基礎であるデジタル理論からスタートし、マイコンの本質を探究、そのマイコンを４ビット、８ビット、12ビットの代表的CPUで製作するとどうなるかを考察。基礎論に基づいて具体的な各メーカのキット作りを実習し、応用製品を開発するための手法など、マイコンに関するすべてを徹底的にマスターでき、しかも、その詳述するテキストは1700頁におよぶ完璧主義の入門。

講座内容

Ⅰ．マイコンの基礎知識
１講　マイコンの基本
２講　ディジタル回路の基礎
３講　マイコンのハードウェア
４講　マイコンと入出力機器インタフェース
５講　マイコンのソフトウエア

Ⅱ．マイコンシステム製作の実際
６講　８ビットCPUを用いたワンボードコンピュータの製作
７講　12ビットCPUを用いた電子楽器の製作
８講　４ビットCPUを用いたワンボードコントローラの製作

Ⅲ．マイコンキットの製作
９講　インテルマイコンキットの製作
10講　日電マイコンキットの製作
11講　モトローラマイコンキットの製作
12講　東芝マイコンキットの製作

Ⅳ．マイコンの開発と応用
13講　マイコン応用システム開発技法
14講　マイコン関連知識
15講　その後のマイコン及び補説

受講料　￥35,000

執筆者
インテル，日電，モトローラ，東芝など第１線活躍者　10名

あなたはどのコースからマイコンを始められますか？

申込：各コース共　月末締切　資料請求は掲載誌名を明記の上、下記へお申込下さい。

マイテック　〒103　東京都中央区日本橋茅場町2-1
☎03—661—3366（代）　市川ビル

東北６県取扱先　**アルプス コンピュータ エンジニアリング**　仙台市一番町２-８-18
〒980　☎0222(66)4447

マイコンの《システム・コンポ》

SWTPC 6800シリーズ

■大量生産でお求め安い価格を実現！
■豊富な周辺装置でシステム構成が容易！
■エディタ、アセンブラ、BASICなど完璧なソフトウエア！
■BASICによるアプリケーションも完備！

●モトローラ6800　8ビット並列マイクロプロセッサ・ファミリーの特長をあますところなく生かしたシステムパーツが作られています。
●どのマイコンにもつなぐことができる互換性のある各種周辺機器と共に完成度の高いコンピュータシステムが製作できます。

米国でトップクラスの実績を誇る

サウスウェスト テクニカル プロダクツ

コンピュータ・システムキット

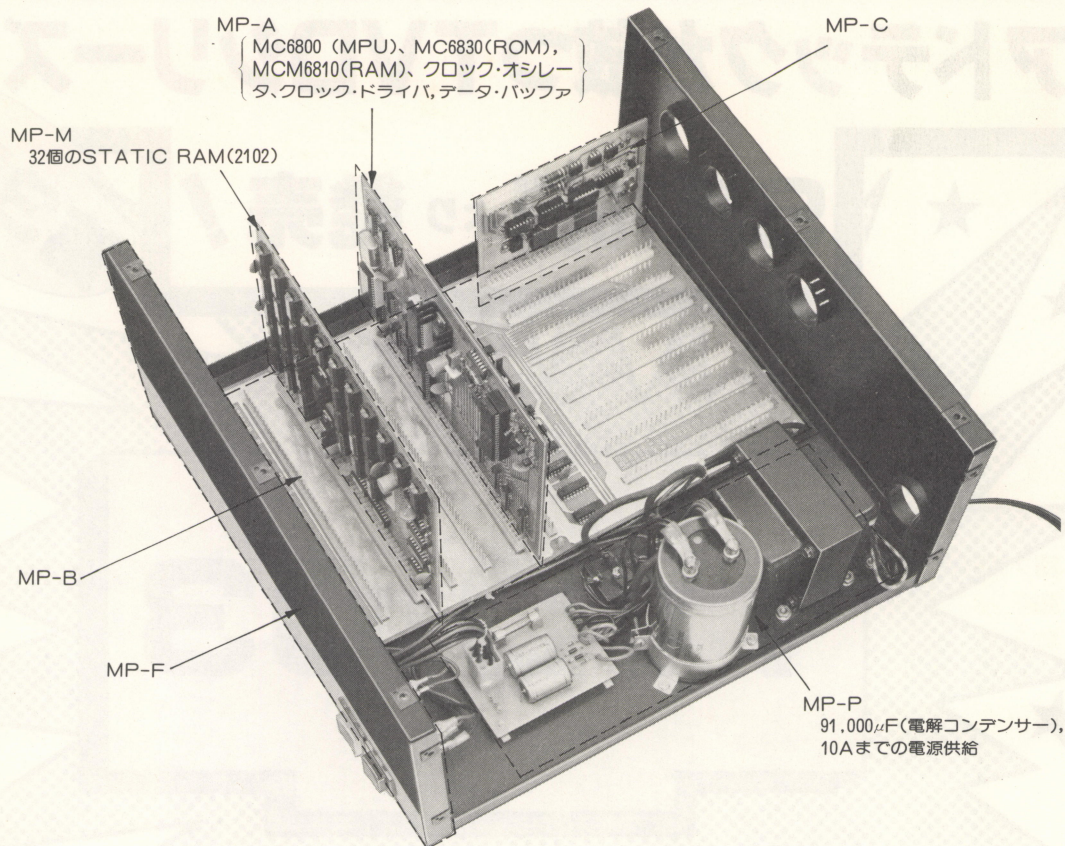
MP-68　マイコンシステムキット
¥198,000　別売可

▶**MP-A　6800コンピュータ・システムボード**　(¥89,000)
ミニオペレーティング・システム用ROM, RAMなどを含む。
▶**MP-B　マザーボード**　(¥25,000)
MPUボード，メモリボード用7スロット，並びにI/O用8ポート付，アドレスコーダなども含む。
▶**MP-C　シリアルインタフェース・コントロール**　(¥25,000)
20mA TTY，RS232ターミナル・インターフェースを含む。
▶**MP-F　シャーシおよびカバー**　(¥18,000)
▶**MP-M　4Kバイト付メモリボード**　(¥49,000)
▶**MP-P　電源ユニット**　(¥21,000)
16Kバイトメモリ，8I/Oインタフェースまで電源供給可。

▶**MP-8M　8Kバイトメモリ**　近日発売予定
▶**MP-S　シリアルインタフェース・ボード**　¥28,000
MC6850ACIA用
▶**MP-L　パラレルインタフェース・ボード**　¥28,000
MC6820PIA使用
▶**MP-T　割込タイマボード**　¥29,000

周　辺　機　器

▶**PR-40　アルファニュメリック・ドットマトリクス・プリンタキット**　¥150,000
5×7ドットマトリクス・プリンタ
ASCIIコードの英数字64種，キャラクタ・ジェネレータ付
40字／行
75行／分(60Hz)，62.5行／分(50Hz)
TTLコンパーチブル・インタフェース，並列インタフェース
FIFO
シャーシ，電源

▶**CT-64　ターミナルキット**　¥195,000
ASCIIエンコーダ付キーボード（56種類，大文字・小文字を含む，128ASCIIセット）
7×9ドットマトリックス（64字／行または32字／行）
×16行／ページ
カーソルコントロール機能，スクローリング機能，リバース機能，ハイライティング機能，ビーパ機能
シリアル・インタフェース(110～1200BPS) TTLコンパーティブル
2頁メモリ付
シャーシ及び美麗ケース付

お求めやすい分割払いの

SWTPCクレジットをご利用下さい

▶MM-I　CRTモニタ(組立済み)　￥107,000
9インチCRTグラフィック・ディスプレイ
完全ソリッドスティト化
美麗ケース付

▶AC-30　オーディオカセット・インタフェース付　￥48,000
カンサスシティスタンダード・インタフェース付
デュアルカセット・ドライブ機能
電源回路
シャーシおよびケース

▶KBD-5　キーボード(56キー)/エンコーダキット　￥30,000
ASKIIエンコーダ，PCボード，オートリピート付

▶GT-61　グラフィック・ターミナルコントローラキット￥59,000
ディスプレイ・パターン　64×96ドットグラフィック・コントロール
パターン・バッファリングメモリ(6144ビット)〔CRTディスプレイ，パラレル・インタフェースおよび電源が必要です〕

▶PPG-J　ジョイスティック・キット　￥24,000
グラフィック・パターンを自由にコントロール
360度回転操作レバー
ポランショメータ／ディジタイザ

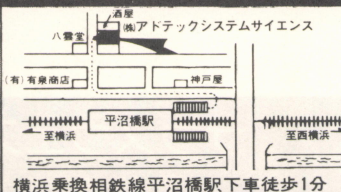
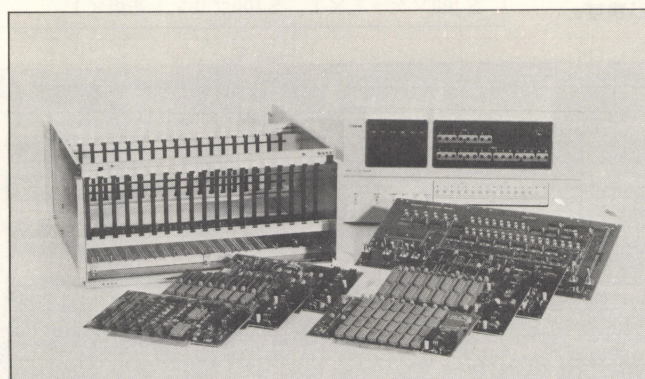
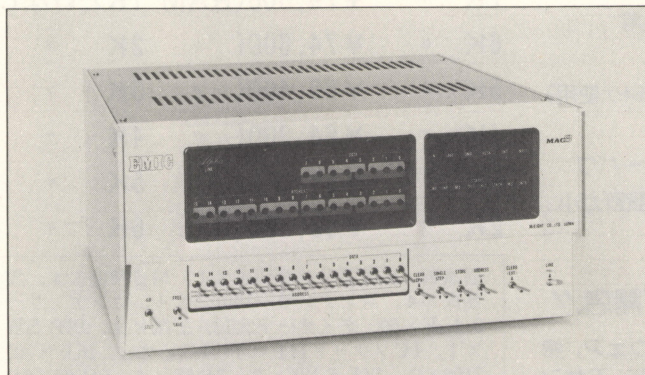
ソフトウェア

▶システム・ソフトウェア
BAS4P：4K　BASIC　言語マニュアルおよび紙テープまたはカセットテープ　￥8,000
BAS8P：8K　BASIC　言語マニュアルおよび紙テープまたはカセットテープ　￥12,000
MP-EP：エディタ／アセンブラパッケージ　￥12,000

▶アプリケーション・ソフトウェア
GAMI：ティクタック・トー，ブラックジャック・ゲームのソフトウェア　￥3,300
ANIM：子供用動物ゲーム　￥3,300
RACE：宇宙船ゲーム　￥3,300

**▶ 101 BASIC　COMPUTER GAMES（英文）
やさしいBASIC（日本語）付**
￥5,000
※今アメリカで大評判の101種類もの面白いゲームのソフトウェアがBASICで書かれ（プログラムのリスト付）て本になっています。

サウスウエスト　テクニカル　プロダクツ　ジャパン（株）

〒150　東京都渋谷区宇田川町2番地1号　渋谷ホームズ518　☎(03)476－0750(代)

マックエイトのマイコンキット EMIC（エミック）シリーズ

価格表

●**本体キット**　完成品￥90,000高

EMIC-Alpha	￥260,000
EMIC-BETA	￥260,000

（基本構成
筐体、電源、パネルインターフェイス
CPU, IKRAMボード）

●**ボードキット**　完成品￥20,000高

CPU-8080	￥40,600
CPU-6800	￥38,700
ZK RAM	￥47,500
4K RAM(2101)	￥77,600
4K RAM(5101)	￥138,000
4K ROM（デバイスなし）	￥25,600
シリアルI/O（TTY）	￥37,200
プログラマブルI/O	￥58,500
アイセレーターI/O	￥50,300

他

●**ボード（結線図付）のみ**

￥15,000	￥15,500
CPU-8080	CPU-6800
シリアルI/O	2K RAM
プログラマブルI/O	4K RAM
アイソレーターI/O	4K ROM
ユニバーサル	￥3,500

取　扱　店（8月1日現在、敬称略五十音順）

アスターインターナショナルCOSMOSチェーン全店
アスター本店では通販も行なっています。(03)354−2661
デンゲル（上野池の端）――――(03)823−4401
日の丸無線通信工業（秋葉原）――――(03)255−2923
ロジックハウス（新宿）――――(03)363−2651
Ⓒ日本信販ローン取扱います。

●I/O次号にて詳細発表！

株式会社 マックエイト
〒222 横浜市港北区樽町511−1
TEL (045)543−1161

ロビン電子産業㈱＆コンピューターラブ㈱
共同製作により、待望の新製品を開発第一弾

ロビン電子

8K Byte Low Power Ram Board Kit

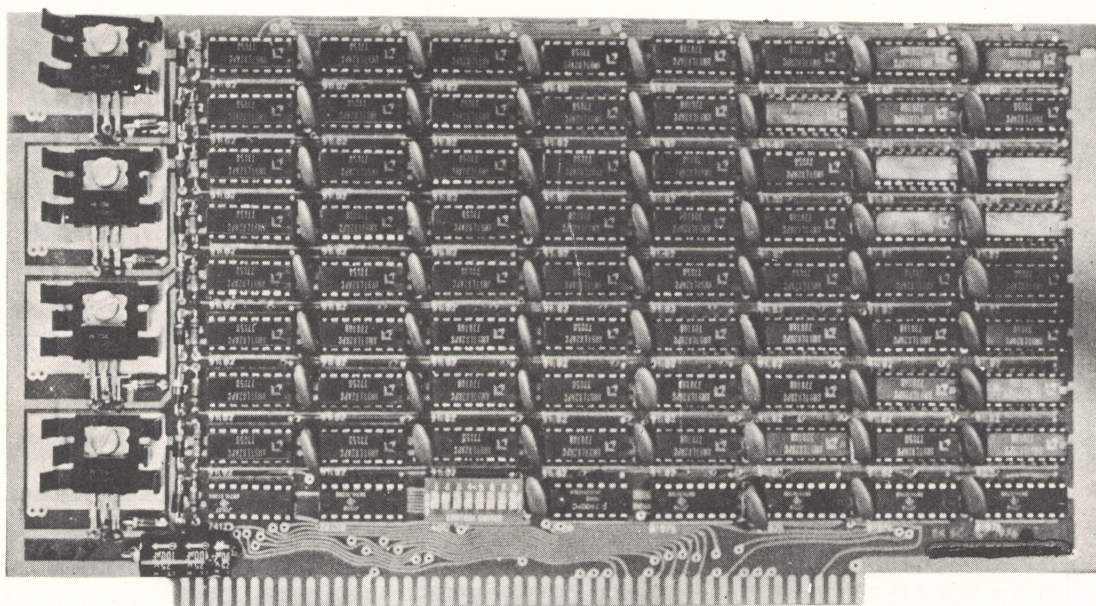
◎8K BYTE 完全キット ￥85,000

◎バッテリーバックアップ可能（8Kバイト600mA）
◎＋8V単一電源（非安定化）又は，5V安定化
◎ALTAIR, IMSAIコンパチブルS100バス用
◎他のCPUにも変更可能
◎ローパワーRAM,1024×1（91L02APC×64ケ使用）500nS
◎スタンダード８Kバイト4.4A（MAX），ローパワー8Kバイト2.6A（MAX），この様に電源設計が小さくてすみます。

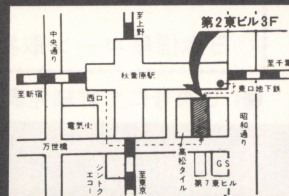
8K BIT	￥85,000	（完全KIT）
7K 〃	￥79,800	（RAM 1Kだけなし）
6K 〃	￥74,600	（〃 2K 〃）
5K 〃	￥69,400	（〃 3K 〃）
4K 〃	￥64,200	（〃 4K 〃）
3K 〃	￥59,000	（〃 5K 〃）
2K 〃	￥53,800	（〃 6K 〃）

部品表，配線図及び説明書×１，MC7805×４，放熱板×４，コンデンサー100μF×１，コンデンサー0.1μF×60，ダイオード×13，抵抗×９，DIP SW×１，ICソケット14P×１，ICソケット16P×70，7400×１，74LS138×２，74365×４，91L02×64，S100ソケット×１，S100プリント基板×１

☆当社２Fにコンピュータラブ開店!!

マイクロコンピュータに関するソフトウェア，特にプログラムの作り方PROMの書込み，アセンブラ，ベーシック等について相談に応じます。

◎ご注文は現金書留・為替にて，住所・氏名・品名・個数・郵便番号をはっきり書いてお願い致します。
●送料：3,000円以下→〒200/3,000円以上→〒500
◎多数お買い上げの方には，別途見積り致します。
地方業者，ユーザー，メーカー大歓迎！

●当店はビル３階のため来店の時は第２あずまビル（10階建）と聞いて下さい。（東口及び地下鉄の方，駅より50mです。）

ロビン電子産業㈱ I/O係
〒101 東京都千代田区神田佐久間町1-14 第二東ビル306号室
TEL.03-255-6028㈹ 営業時間／9：30～19：00 休日／日曜日

★第２弾 テープリーダー近日発売 乞う御期待!!
★３F移転記念 在庫大処分好評実施中!!

独学のための通信講座 6ヶ月コース（期間9月30日～53年3月）

マイクロコンピュータ技術スクール

4ビット編初級コース(自作と応用)

電子技術の知識が特別になくても誰にもわかりやすい解説と構成
簡単な自動化と機械・装置へ応用するには4ビットが最も安価です。

昭和52年9月開講 第8回生募集ご案内

講師 杉田 稔氏
（杉田技術研究所・所長）

この通信講座の修得方法

1. 最初1回目のテキストと一緒に講師著「実用マイクロコンピュータ」￥2,800を無料で提供し、基礎的知識を修得していただきます。
2. 受講者は随時質問を講師に提出し、適当な時期に解答が得られます。
3. テキスト学習だけでなく、添削による指導(2回)、全カリキュラム終了後のスクーリング(1日)を実施します。
4. 毎月のテキストに設問があり、その模範答案が次回のテキストに発表されております。

回	内容	回	内容
第1回テキスト	●マイクロコンピュータはなぜ必要か●マイクロコンピュータを学ぶには●マイクロコンピュータの独学計画●マイクロコンピュータはどのくらいの費用，時間がかかるか	第5回テキスト	●マイクロコンピュータの自作計画●自作上の要点，注意●マイクロコンピュータの自作方法●マイクロコンピュータの全回路の解説●自作時の部分的計測方法
第2回テキスト	●ディジタルと2進級●ハンダ付けと配線方法●マイクロコンピュータとは●マイクロコンピュータ自作に必要なもの●電源について●回路図の見方，回路について●マイクロコンピュータ自作の注意●テスタの使い方●各素子の扱い方●マイクロコンピュータはどんなところに使うか●TTLとトランジスタ●基本回路の実験方法	第6回テキスト	●命令について●命令の解説●命令の種類と解説●命令についての参考資料●基本プログラムの解説●簡易プログラムで自作コンピュータを動作させてみんる●簡易プログラム解説●その他
第3回テキスト	●TTLとは何か●TTLの使い方●C-MOSの使い方●オシロスコープの使い方●フリップフロップ●メモリ●シフトレジスタ●カウンタ●ラッチ●デコーダ●その他のTTL●ディスプレー●マイクロコンピュータとインタフェース●マイクロコンピュータの入出力の回路	第7回テキスト	●マイクロコンピュータの電源接続●マイクロコンピュータのまとめ●コンピュータの入力技術●コンピュータの出力技術●機械・装置その他との入力方法●入力・出力インタ・フェース●RAM出力の応用技術●コンピュータ応用技術●質問事例の解答
第4回テキスト	●マイクロコンピュータの構成●電源部分の自作その他●マイクロコンピュータ用各素子●CPUについて●RAM 4002について●ROM4001について●P-ROM4702について●インターフェイス4289について●RAM2101について●クロックゼネレータについて●シフトレジスタについて●SN74157について●RAM2102について●RAMでプログラムメモリを行う●マイクロコンピュータの動作解説	第8回テキスト	●新4ビットマイクロコンピュータ回路図●新回路図の解説●システム開発のための技術●参考資料●その他
6ヵ月目	スクーリング 1日（日時・場所、受講者に連絡）		●マイクロコンピュータ自作の要点●マイクロコンピュータ着想・計画・応用の要点●出力部の計画・出力部の実際●入力部の計画・入力部の実際●インターフェース(TTL，トランジスタその他の応用回路)●トラブル対策●マイクロコンピュータの経済計算●質疑応答

※マイクロコンピュータ自作用キットを販売します（詳細は受講者に連絡）

この通信講座の特色

1. 独学で実力がつくよう、きわめて平易な独特の解説をします。
2. 講師の製作応用の体験からわかりやすく詳説します。
3. マイクロコンピュータ応用の体系的な実力をつけるカリキュラムを配慮しました。
4. 受講者すべてが、自作応用できるように指導します。
5. 質問票をご利用になりスクーリングにご参加することによって次のようなメリットがあります。

◎マイクロコンピュータに関する新しい情報の提供。
◎マイクロコンピュータ自作上の相談。
◎自作したマイクロコンピュータの診断。
◎自作上のトラブルの解決とその指導。
◎計測応用技術のための相談と指導。

受講要項

期間 昭和52年9月30日～53年3月

受講料

1名につき	38,000円
3名以上	35,000円
5名以上	32,000円
10名以上	29,000円

※受講料の中には、「実用マイクロコンピュータ」講師著（￥2,800）テキスト6冊，添削，スクーリングなどすべての費用を含みます。

下記申込書をお送りください。

受講料は、現金書留，銀行振込　住友・新宿（当）　三菱・新宿（普）　富士・新宿（普）　三和・新宿（普）

着次第、領収書、受講証をお送りします。

キ…リ…ト…リ…線

受講申込書 I/O 係

会社(工場)名，個人の場合は個人名		電話	会社 / 自宅
所在地	□□□-□□		
申込担当課	部課	申込者	

所属	氏名	所属	氏名

主催／ 新技術開発センター

東京都新宿区三光町1　花園ビル（伊勢丹新館前）
電話　東京（03）209－9661（代）〒160

6ヵ月でマスターできる通信講座（期間52年9月30日～53年3月）

8ビット新製品8085による実用コース

講師が実験に実験を重ねたその体験をテキストに再現！

昭和52年9月開講 第1回生募集案内

講師 杉田 稔氏
（杉田技術研究所・所長）

この通信講座の修得方法

1.最初1回目のテキストと一緒に「マイクロコンピュータ活用事典」¥1,800を無料で提供し、基礎的知識を修得する準備をしていただきます。
2.その後、毎月定期的にテキストを配布します。(裏面参照)
テキストと一緒に質問用紙が添付されており、受講者は随時質問を提出し、適当な時期に解答が得られます。
3.テキスト学習だけでなく、添削による指導(2回)、全カリキュラム終了後のスクーリング(1日)を実施します。
4．毎月のテキストに設問があり、その模範答案を次回のテキスト配布の際に発表いたします。

第1回テキスト	●8ビットマイクロコンピュータを学ぶには●4ビットと8ビットの違い●8ビットマイクロコンピュータでどんなことができるか●独学の方法●何を学べばよいか●独学時間はどのくらいかかるか●費用はどのくらいかかるか●8ビットマイクロコンピュータ製作に必要な回路部品の実用的な解説●その他
第2回テキスト	●電圧と電流の解説●回路部品の実用解説（抵抗器、コンデンサ、トランジスタ、TTL、電線、スイッチ、コネクタ）●テスタの使い方、はんだ付け方法●電源の解説●回路図の見方と配線方法●オシロスコープの使い方●8ビットマイクロコンピュータとはどんなものか●その他
第3回テキスト	●1か0の話し●ディジタルと2進数●ディジタルな入門実験と解説●2進数でなんでも表現する●ICの話し●TTLの実験と解説（記憶とはどんなことか、計数とは、ゲートとは、その他）●CMOSの実験と解説●8ビットマイクロコンピュータに必要な基礎技術解説●8ビットマイクロコンピュータ用LSIの各種解説●LSIの取扱い●8ビットマイクロコンピュータの構成●その他
第4回テキスト	●8ビットマイクロコンピュータの内部的解説●8ビットマイクロコンピュータの動作解説●8ビットマイクロコンピュータの回路図●8ビットマイクロコンピュータの自作方法●配線方法、取扱い方法、電源の接続●入力と出力について●自作マイクロコンピュータの回路点検方法●自作マイクロコンピュータに役立つ資料●その他
第5回テキスト	●自作マイクロコンピュータ試運転とその注意●8ビットマイクロコンピュータとプログラムについて●命令とコンピュータの動作解説●命令解説●まずフローチャートの作成方法●プログラムの作成について●簡単な基礎命令で先ず動かしてみよう●簡易プログラムで動かしてみよう●プログラムの幾つかの解説●その他
第6回テキスト	●8ビットマイクロコンピュータの入力と出力にどう接続するか●各種接続の技術●8ビットマイクロコンピュータと機械の関係●8ビットマイクロコンピュータで機械的パワーを得る技術●入力出力のインタフェース●各種検出器と入力方法●A－D変換器とその実験解説●D－A変換器とその実験解説●ノイズの注意●8ビットマイクロコンピュータの応用各種解説●その他
6ヵ月目	**スクーリング 1日**（日時・場所、受講者に連絡）●マイクロコンピュータ応用の経済計算●マイクロコンピュータのまとめ●マイクロコンピュータ2～3台による各種実演とその技術ノウハウ内容の解説●質疑応答

※マイクロコンピュータ自作用キットを販売します（詳細は受講者に連絡）

この通信講座の特色

1.従来からの通信講座の経験を活かして、受験生からの質問、希望などを多く取り入れ、8ビットマイクロコンピュータが独学で自作できるように一新した内容です。
2.講師自身が、8ビットマイクロコンピュータを製作応用した体験からわかりやすく詳説します。
3.8ビットマイクロコンピュータ応用の体系的な実力をつけるため次のようなカリキュラムを配慮しました。
○8ビットマイクロコンピュータの機構
○8ビットマイクロコンピュータの自作
○プログラミングの基礎
○マイクロコンピュータのソフトウエア
○8ビットマイクロコンピュータの周辺機器
4.受講者すべてが、自作、応用できるように指導します。

受講要領

期間 昭和52年9月30日～53年3月
受講料 1名につき 38,000円
3名以上 35,000円
5名以上 32,000円
10名以上 29,000円
※受講料の中には、「マイクロコンピュータ活用事典」（¥1,800）テキスト6冊、添削、スクーリングなど、すべての費用を含みます。

下記申込書をお送りください。
受講料は、現金書留、銀行振込 ｛住友・新宿（当） 三菱・新宿（普） 富士・新宿（普） 三和・新宿（普）
着次第、領収書、受講証をお送りします。

キ…リ…ト…リ…線

受講申込書I/O係

会社(工場)名，個人の場合は個人名		電話	会社 / 自宅
所在地	□□□-□□		
申込担当課	部課	申込者	
所属	氏名	所属	氏名

キリトリ線

主催／ 新技術開発センター

東京都新宿区三光町1 花園ビル(伊勢丹新館前)
電話 東京(03)209－9661(代) 〒160

国際派のキミのための

工業英語講座

連載

ビールを飲んで工業英語を勉強しよう

榊原祐輔

ビールのうまい季節ですネ．今回は，偶然手に入れた，アメリカのビールに，次のような文が印刷されていたので，これについて少し考えてきましょう．

①Our famous artesian water enables us to attain perfection in art of brewing

工業英語には，この『enable＋目的語＋to不定詞』のパターンが多くでてきます．前回にのべたallow, causeも同様のパターンをとります．ほかにもpermit, make, letがありますが，makeとletは，toを少略した，不定詞が続きます．

例をあげて，詳しく説明していきましょう．これはあるマニュアルからとったものです．

②This enables the processor to resume excution of the main program.

③This instruction allows the programmer to alter the contents of the resister.

④This causes the processor to resume excution of the program.

enable, permit, cause, allow は同様に使用されますが，若干の差異はあります．enableはmake possibleの意味で，目的語には①の例のように，人称代名詞又は，人に関した名詞が続くことが多いようです．しかし，②に示されるように，人以外の目的語も当然とることができます．つまり人以外の目的語の場合，擬人化したと考えればよいのです．

②の文の，allowはpermitにおきかえられます．両方とも『許す』の意味ですが，allowが『妨げる意志がない』『かまわず〜させておく』という negative（消極的）な許すであるのに対して，permitは，『同意や許可を与える』という positive（積極的）な，意志を表現します．

工業英語では，主語に非生物がくる場合が多いので，そのものに，意志などあるわけありませんが，すべて擬人化したと考えればよいでしょう．

④は，原因または要因と結果を明確に述べたいときに使われます．

②，③，④の文は，make, let を使っておきかえることが可能ですが，幼稚な文になります．特にletは口語調で，数式などを仮定する場合をのぞいて，工業英語では，ほとんど出てきません．

④を make を使っておきかえてみます．

⑤This makes the processor resume excution of the program.

不定詞の　toがとれることに注意しましょう．

makeはあいまいな動詞なので，causeを使った方が，工業英語らしくなります．resumeには，メイン・プログラムなどが，インタラプトなどの要因によって一時中断し，再びプログラムを開始するというニュアンスがあります．resumeをrestartなどのやさしい英語におきかえることが，可能ですが，意味が少しぼけてきます．

⑥This makes the processor restart (the) program excution.

またはexcutionを名詞から動詞にして

⑦This makes the processor excute the program again.

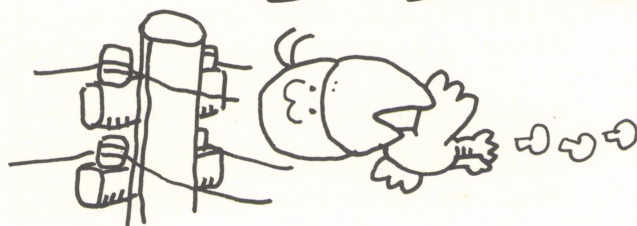
⑥でわかるように，excution of the programでもよいのですが，(the) program excutionとした方のが，簡潔な表現になります．この場合，前置詞のofを使うかinを使うかなどという余計な心配がなくなります．また，定冠詞のtheをつければ，特定のプログラムの実行になりますし，theをとれば，実行という全体の概念を示すことになります．　⑦のように，工業英語では，動作を名詞形で書いてあることが多いので，動詞にもどすことができます．

③を訳してみます．『この命令によって，プログラマーは，レジスターの内容を変えることができます．』ここでallowを『〜によって』と訳すとピンとくると思います．逆に『〜によって』または『〜を使えば』という表現は，『by using〜』などとしないで，このallowという文型を利用すると，英語らしい表現になります．ついでにalterという，単語も覚えておきましょう．changeしか知らないのでは，困ります．①の文にもどってみましょう．

訳はいろいろ考えられると思います．

『当社の（当地の…）上等の掘抜き井戸水を使用しているので，ビール醸造技術は，最高にすばらしいものです．』

Ourは，ビールを醸造している会社，またはその地域と考えられます．会社は1人では，成りたたないので複数形です．famousは有名ということですが，とびきり上等というニュアンスを含んでおり，excellent, first-rateといった言葉におきかえてもいいでしょう．perfectionには成熟という意味がありbrewingにひっかけていますが，この場合，完全というよりは，比類するものがないほど最高級というニュアンスがあります．またartをtechniqueにしなかったことによってこの文が生きてビールの芸術品というニュアンスが加わります．

シンセサイザのつくり方

岩崎　弘

1 ミュージック・シンセサイザとは何か

ミュージック・シンセサイザ（音楽合成器）が，自作マニアの間で話題になり始めたのは，ほんの1～2年前のことでした．最近では，マイコンのアマチュアへの普及によって，マイコンでの自動演奏へと応用も広がりつつあります．

音を電子的に合成するためには，数多くの方法が試みられ実用化されてきました．その代表的な商品が電子オルガンです．これはあらかじめ合成された音を演奏者が再び選択，合成するという方法で演奏されます．

これに対して，ミュージック・シンセサイザは，音の素材をよりナマの形で表面に持ち出し，演奏者の手にゆだねています．したがって，電子的に音を合成するにはどうしたらよいかのアウトラインを知っておくことは，演奏する場合やマイコンで自動演奏させる場合に重要になるわけです．

★音の3要素

音の3要素とは，**強さ**，**高さ**，**音色**の事を言います．つまり，強さとは音の波の高さ，高さとは音の波の周波数，音色とは音の波の波形の事を言います．しかし，実際には，3つの要素がある意味で渾然一体となり，広義の音色というものを形成しています．

例1) 同一の波形の波を時間とともに変化させると音色が異なる．**図1**にその変化と音色を示します．

例2) 同一の波形の波の高さを変えると音色が違って聞こえる．たとえば，レコードなどの回転数を変えて聞くと違った音色に聞こえます．

例3) トランペット音の立ち上がり部分の音のみ消去すると，フルートなどと聞分けがつかない音になります．

つまり音は，単純に3つの独立した要素から成り立っていると考えるよりも，波形あるいは部分音の含みぐあいの時間的変化により，音色が形成されると考えられます．ミュージック・シンセサイザとは，音の3つの要素を，電子的にいろいろ変化させ，音楽を合成するものと言えます．

2 シンセサイザの基本構成とその機能

シンセサイザの基本構成は，**図2**で表わせます．また，各ブロックにどのような機能が含まれているかを，**表1**に示します．

音源は音色形成器を通って出力となり，双方とも手動および自動コントローラで時間的に制御されます．演奏するということは，手動コントローラを調整することであり，それによって直接音源の発振周波数を決め，音色を選び，あるいは自動コントローラの各定数を定め，間接的に音源，音色形成器を制御するわけです．

音源としては，**表1**に示すいろいろな波形のものがあり，これを手動および自動コントローラで，その発振周波数を決定します．音色形成器は，音源信号を**表1**に示す機能により，いろいろな波形に変化させます．自動コントローラは，演奏者が一度命令を与えると，あらかじめ決定されている手順にしたがって制御電圧を発生するものです．手動コントローラは，演奏者と

I/O プラザ　▶梅雨でジメジメ，晴れたと思うとカンカン照り，汗の出ない日はありませんが，I/Oはなかなか充実してると思います（日本の現状……僕も含ム！をよく反映しているのだけれども）．
なにせ8KBのメモリーをどうこうしようとしても，ボードから作らにゃならんのだ，誰でもいいからバスの標準を決めて，メモリーボードを安く買えるようにしようではないの！　↗

図1　変化と音色

図2 シンセサイザー基本様式ブロック図

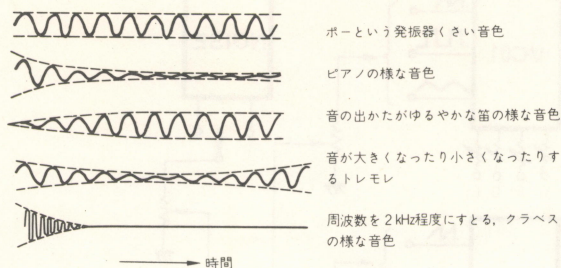



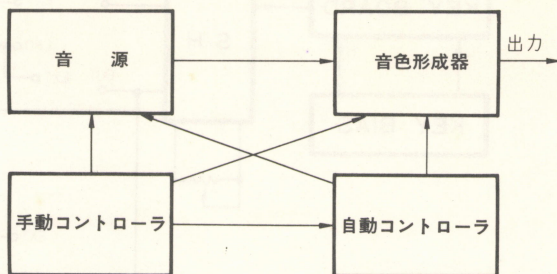
表1　各ブロックに含まれている機能

音　源	正弦波，三角波，方形波，パルス波，およびそれらの二重，三重の合成波，ホワイトノルズ，ピンクノイズ，その他
音色形成器	ローパスフィルタ，ハイパスフィルタ，バンドパスフィルタ，リングモジュレータ，ゲート，固定フィルタ，ミキサー，残響装置，その他
自動コントローラ	発振器，フィルタ，ゲート用包絡線発生器，シーケンサ，変調用各種波形発振器サンプルアンドホールド，その他
手動コントローラ	通常鍵盤，スライドコントローラ，各種切換スイッチ，各種ボリュームコントロールつまみ，タッチセンサー，フットペダル，エクスプレッションペダル，フットスイッチ，他，人間が直接触れるもの

楽器が結びつく唯一の接点です．**表1**にその種類が示されています．

3 シンセサイザーの製作

今回，私が製作したキットの仕様について示します．

〔SK—307〕* の仕様

①44KEYキーボード（木製ケース付）
②コントロールパネル部（シルク印刷および穴加工付金属製パネル，VR，SW，木製ケース付）
③シンセサイザ回路部（VCO×2，VCF，VCA，ノイズソース，AR，ADSR，S/H，電源などが1ボードにはいっている．またPCボードには，シルク印刷がされています．）

上記以外に，オプションとして，LFOキット（SK-308），およびリバーブレーション・コントロールシステム（SK-303）が用意されているので，あとから付けたせばより多くの効果が楽しめるでしょう．なお，パネル部には二つのオプションの制御部が付いています．（＊伸光ミュージック・シンセサイザ・システム・キット）

図3に，このキットのブロック図を示します．

■キーボード部

キーボード部のケースは組み立てられているので，音階抵抗を取り付けるだけです．キーボードの**写真1**と，接点部の**写真2**を示します．

木製ケースは，塗装がされていないので，自分の好みに合わせて，ラッカーなどで塗装すれば見映えも良くなるでしょう．

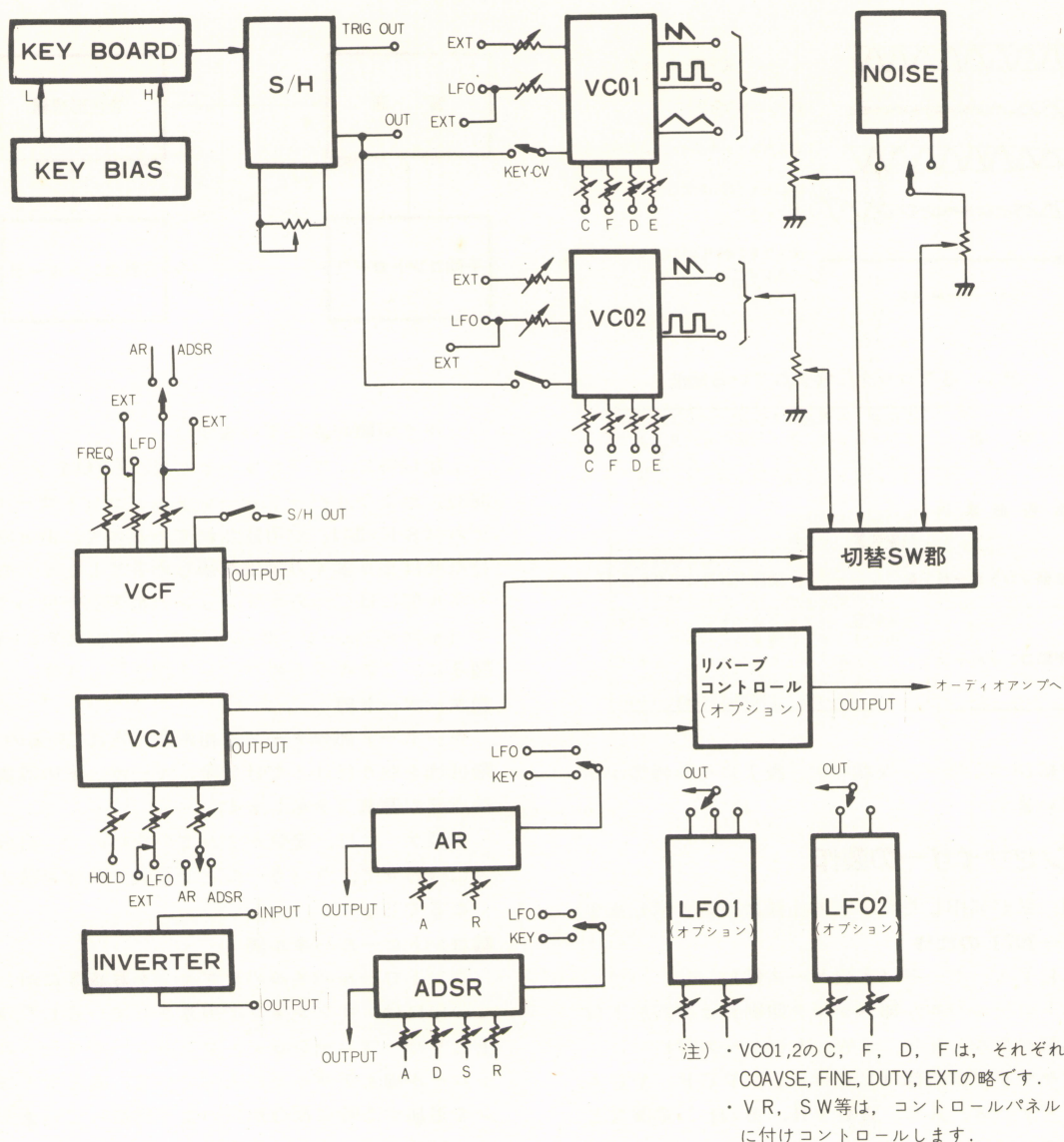
■コントロールパネル部

コントロールパネルのデザインを**写真3**に示します．この中には，オプションで用意されているLFOキットとリバーブレーション・コントロールキットのコントロール部もデザインに入っていて，あとでオプションを追加する時に都合が良いようになっています．また，VR，ロータリーSWなどが各ブロックごとに付

写真1　キーボードの写真

写真2　接点部

メモリーは昔に比べて安くなったのに（1/10以下だよ），その安いメモリーを載せることが大変なんだとは……こんな状態でブームをあおるなんていうのは罪だよ．（I/Oのコトデワナイノダ）
いや，こんな事を書くのではなく，汗まみれになって働いている（遊んでる？）編集部の皆さんに，ゴクローサマと言いたかったのだった．
（西条市　島　敏記）

図3 SK-307のブロック図

いており，かなり自由に合成できるよう設計されています．

製作は，VR，SWなどをまず取り付け，次にマニュアルにしたがって配線してゆきます．シンセサイザ回路部との接続線には，番号などを付けて判りやすくしておくと，あとで楽です．あとは，キーボード部と同様に塗装してケースを組み立てるだけです．のちほどシンセサイザ回路部を中に組み込むので，ケースの裏は組み立てずにしておくと良いでしょう．

■シンセサイザ回路部

シルク印刷がしてあるので，製作はマニュアル通りに組み込んでゆけば誰にでもできますが，各ブロックごとに確実に組み立ててゆく事が完成への近道です．**写真4**に組み上げたPCボードを示します．

ⓐ音源回路部

このキットの音源には，二つのVCOとノイズソースが使われています．VCOは，VCO-Iとして方形波，のこぎり波，三角波が出力できるものとVCO-IIとして方形波，のこぎり波が出力できるものとがあります．ノイズソースとしては，ホワイトとピンクの二つのノイズが出力できます．

次に各種の音源波形の性質とその使用法を述べます．各種の波形とその倍音構造を**図4**に示します．

正弦波……倍音をまったく含まない純音の波で，音合成の基本となります．この音は，音さから出る音と同じように無色透明です．滑らかに上下する性質はコ

I/O プラザ

▶毎号楽しく読ませていただいております，創刊号と次の号は残念ながら購入できなかったのですが，3号以降はずっと愛読しております．
　当方，すでにガラクタの寄せあつめで，一応8080のマイクロコンピュータを製作したのですが，何分，ソフトでコントロールされるパネルスイッチとLEDだけで，I/Oには特に興味があります．　（大阪　山口則正）

写真3　コントロールパネルのデザイン

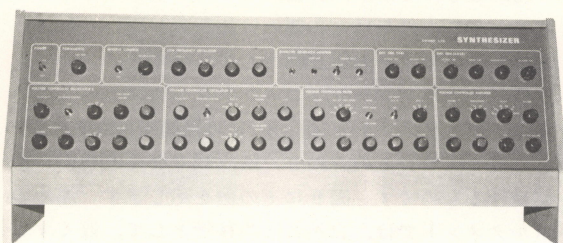



写真4　組み上げたＰＣボード

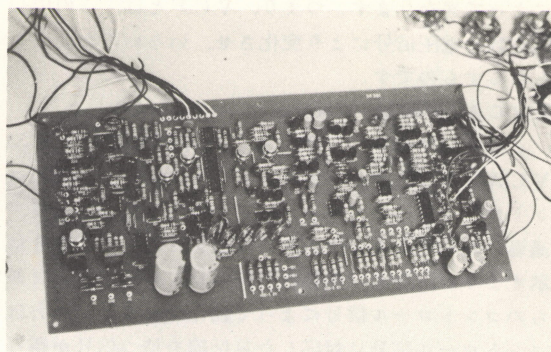
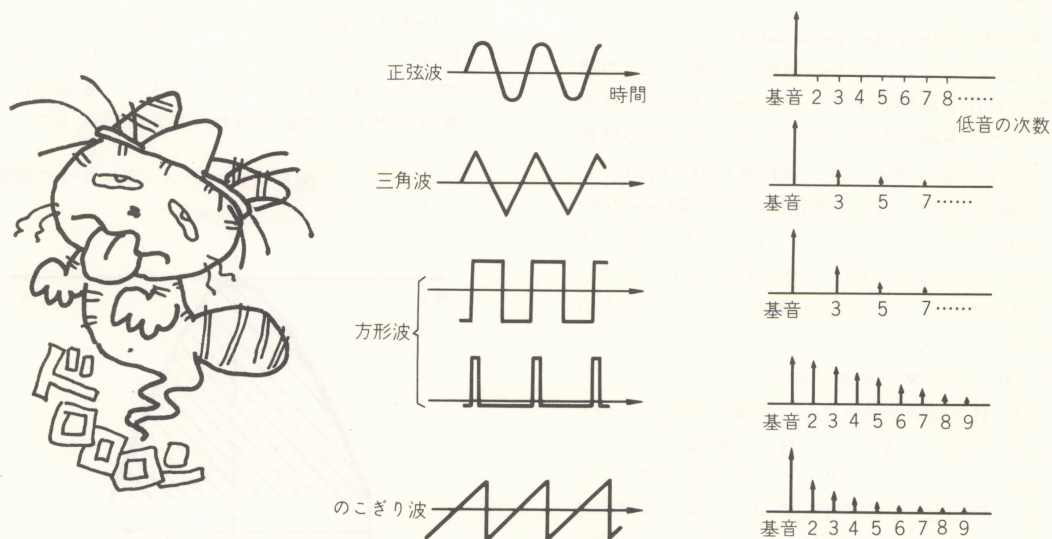



図4 各波形の倍音構造

ントロール信号波形としても利用価値があります．たとえば，音源発振器の変調用に数Hzの正弦波を使用するとビブラートに，振幅を正弦波で変調するとトレモロになります．これはオプションのＬＦＯキットに入っています．

三角波……頭のとがった三角の波で，そのスロープは直線です．少量の奇数倍音（3, 5, 7, 9, 倍音…）を含みます．フイルタ操作により多少音色が変えられ，コントロール信号としても有用です．

方形波……２つの電圧を交互にとる波形で，山と谷の幅が同じであれば，その倍音構成は奇数倍音のみです．これは，クラリネットなど閉管の音を作り出すのに適しています．また，コントロール信号として使用すると，音の断続，リピート操作などができます．さらに音源をこの波形でＦＭ変調すると，異なるピッチの音を交互に出すことができます．

のこぎり波……倍音列の次数を倍音の大きさが反比例しています．きれいな倍音構造のため利用価値もかなり高い．この波形の音色は明るい擦弦楽器の感じですが，フィルタを通すことにより全管，リード楽器，その他各種の音色を作り出すことができます．

雑音源……特定の周期，波高を持たない不規則な波形で，音源としてもコントロール信号としても使えます．波，雷，風などの擬音や音楽中の雑音成分の合成に用いられます．リズムボックスなどに使用されているマラカス，スネアドラム，シンバルなどの音源は雑音を利用しています．ホワイトノイズとは，同波数軸上，同じ帯域幅内に含まれる雑音のことで，ピンクノイズとは，帯域幅オクターブ中に含まれるエネルギーが同じ雑音の事です．コントロール源としては，電圧値の変動をそのまま制御電圧として使う方法や，一度サンプリングしてから使用する方法などがあります．

ⓑＶＣＦ（Voltage Controlled Filter）

このキットのＶＣＦには，ＢＰＦ（バンドパスフィルタ）が使用されています．**図5**にその特性を示します．中心周波数の近くの部分音のみ通過させるフィル

I/O プラザ　7月29，30日に広島大学内でマイコンの入門講習会が開かれました．さらに30日午後，広島近郊のマイコン利用者の初の集いが同大学内でありました．呼びかけが大学研究室であったので学校関係者が主でしたが，学術的？な話はなくもっぱら懇話会．今後も続けられますが，ホビーは対象にならない様子．（広島　タネモリ）

タです．このフィルタを通過した音は，中心周波数fcによって異なります．つまり，ＶＣＦとは，この中心周波数を電圧信号により変化させ，いろいろな効果をもたらすものです．

ⓒＶＣＡ（Voltage Controlled Attenuator）

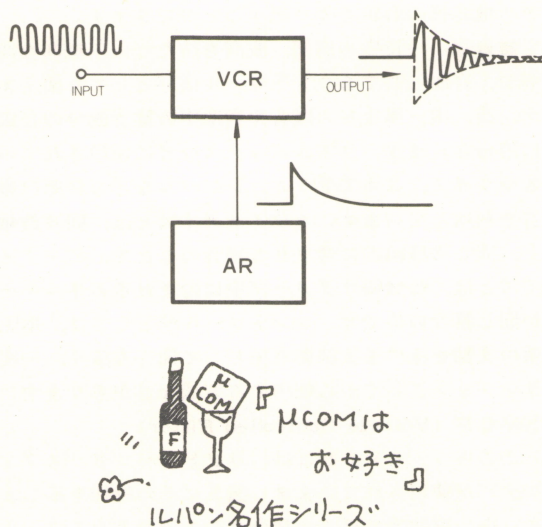
このキットのＶＣＡは，ＡＣ入力換算器型です．ＶＣＡとは，音源またはコントロール信号を，他のコントロール信号に対応して通過させたり閉止したり，通過させる度合を変化させる機能を持ちます．**図６**に示すように，音源に正弦波を使用し，ＶＣＡをＡＲからのコントロール信号によって動作させると，出力はコントロール信号に対応した包括線を持つ信号が得られます．ピアノなどの減衰音，管楽器の定常音などの音が作れます．

ⓓAR，ADSR（envelope generator）

音の分析は，音をいくつかの情報に分解し，その情報の時間的変化を観察します．合成時には逆に，その情報によって音を構成させるわけです．その時の情報を電圧の変化に置き換えた時，包絡線(envelope)といい，その包絡線が発生するものを包絡線発生器(envelope generator)といいます．このキットには，ARとADSRの機能が入っています．**図７**にその特性を示します．シンセサイザでは，この波形を調整し，いろいろな音を作り出すわけです．

ⓔＳ／Ｈ（サンプルアンドホールド）

図８にＳ／Ｈと他のブロックとの接続を示し，そのサンプリングの例を示します．Ｓ／Ｈとは鍵盤からのコントロール信号，ノイズ，ＬＦＯなどの信号をサンプリングして，ＶＣＯ，ＶＣＦなどにコントロール信号を与えます．このキットのＳ／Ｈはポルタメントの調整ができるようになっています．

図６　ＶＣＡ

ⓕシンセサイザのその他の機能

ⓐ～ⓔで述べた機能は，シンセサイザに使用される一般的なものですが，音色によりバラエティを持たせるため，そのほかにも各種の付加装置が用いられています．ここではその代表例として，ポルタメント，ＬＦＯ，リバーブレーションコントロールについて述べます．ポルタメントとは，Ｓ／Ｈの機能中に含まれます．ある音と次の音との音程を，階段状でなく連続的に結びつけるものです．**図９**にその例を示します．速めのポルタメントでは，自然音階の移行として，遅くするとサイレンのような効果も得られます．

ＬＦＯとは，Low Frequency Oscillator の略で，ビブラート，トレモレ，その他音源そのものでなくコントロール電圧として用いる0.1～20Hz程度の発振器を言います．**図10**にその応用例を示します．このキットには，オプションとして，ＬＦＯが付けられます．このオプションは，ＬＦＯ-Ｉとして，正弦波，方形波，三角波が出力できるものと，ＬＦＯ-IIとして，方形波

図５　バンドパス・フィルタの特性

図７　包絡線発生器の特性

I/O プラザ

▶Ｉ／Ｏは第２巻第１号より読んでいます．我会は今のところ実行一歩前ですが，早くもＴＫ－80派，ＳＣ/ＭＰ派，ＴＴＬ派(？)etc……に分かれております，８月号ではＳＣ/ＭＰ派が狂喜しています．付録のＮＩＢＬを見つけてＳＣ／ＭＰ派が狂喜しています，以下ＳＣ／ＭＰ派から……

今度，同じＮＳのMM57109についての記事を載せてください．宮永先生，MM57109とＳＣ/ＭＰを抱き合わせ↗

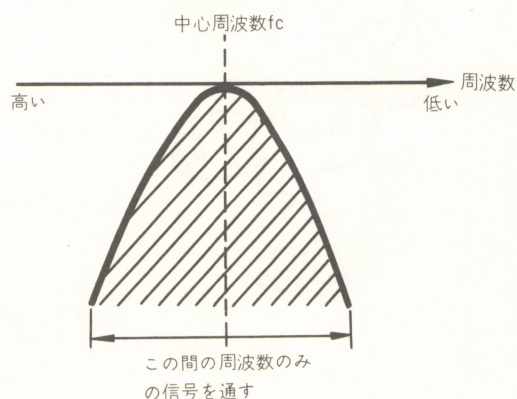
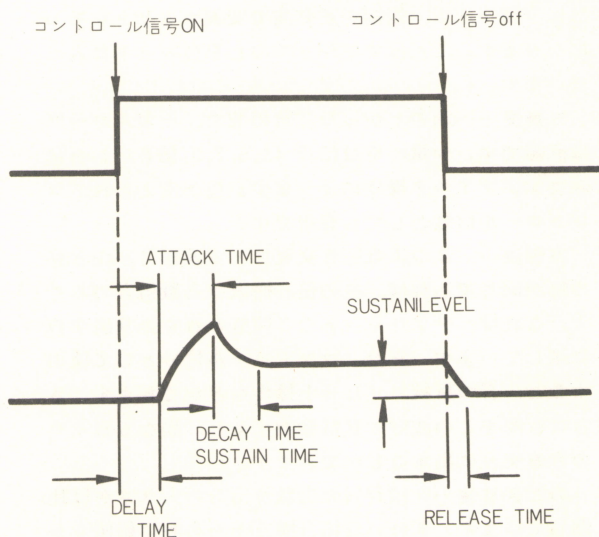



図8　サンプルアンドホールド

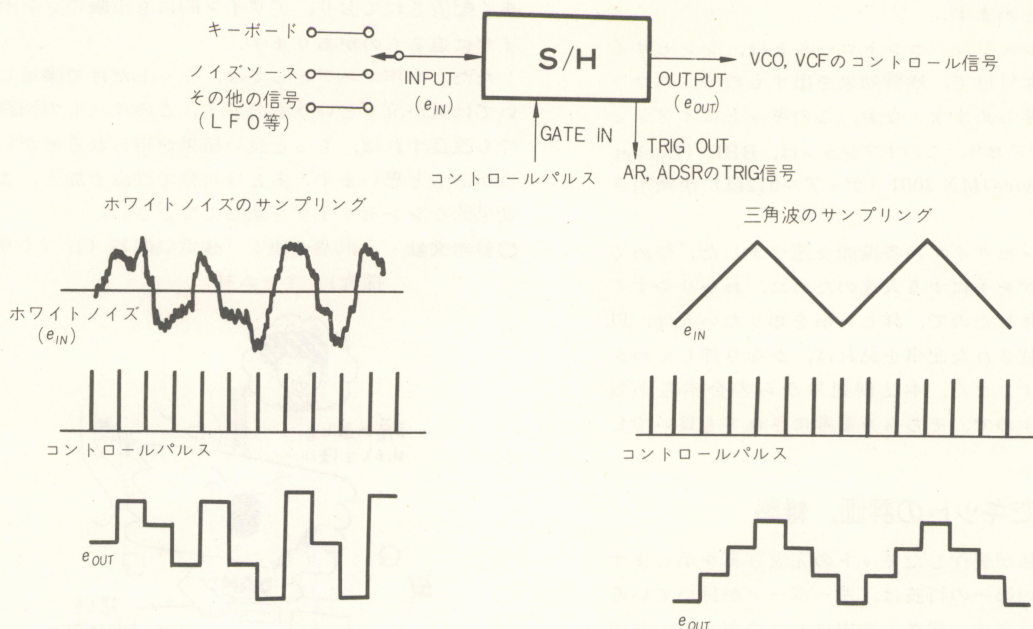



図9　ポルタメント

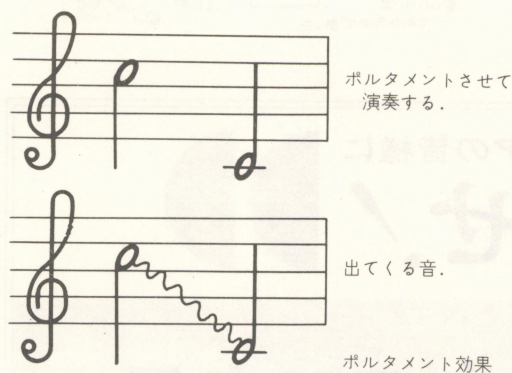



写真5　完成したキット

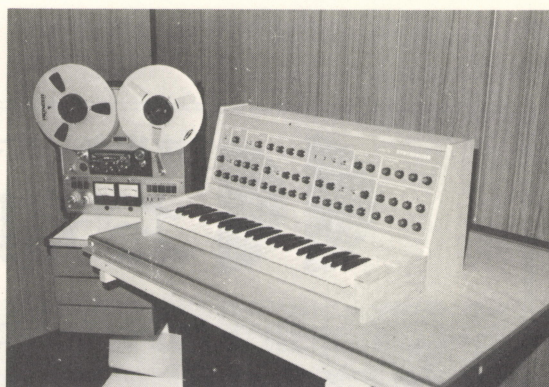
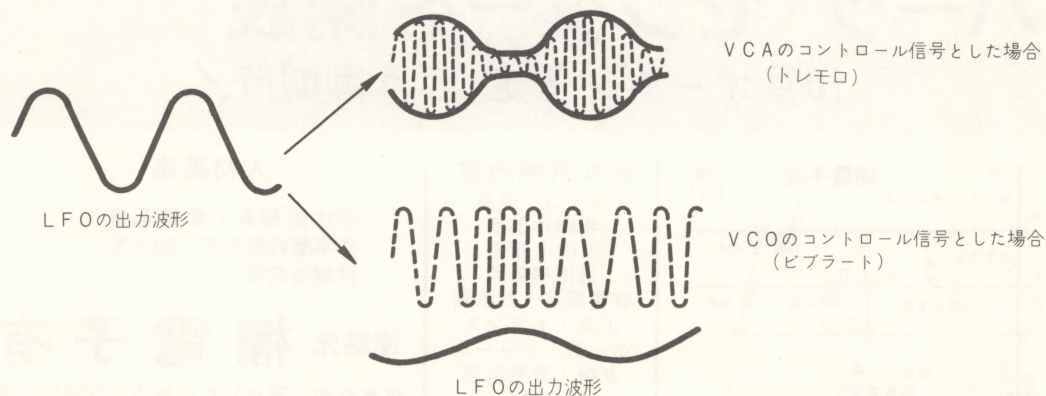



図10　LFO

LFOの正弦波出力をVCA, VCOのコントロール信号とした場合の効果

て，FORTRAN level　7000くらいを走らせられる記事を載せて他のマイコンに差をつけてください．万（よろず）集団，E.S.P，斗荷怒慈慰は千葉県にて結成された集団で，現在3人の大学3年生により構成されています．電算分科会には2.5人(？)が加入しています．（万集団　E.S.P　斗荷怒慈慰　電算分科会）

とのこぎり波の出力できるものが入っていて，かなりの効果が楽しめます．

リバーブレーションコントロールとは，シンセサイザの出力部に付けて，残響効果を出すもので，エコー効果なども楽しめます．なお，このキットにオプションとして付いており，このオプションは，BBD（Bucket Brigade Device）MN 3001（デュアル512段）が使用されています．

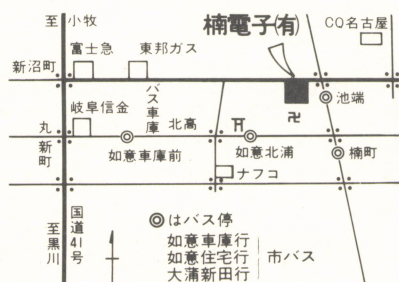
以上，シンセサイザの各機能を述べました．初めてシンセサイザを手にする人達のために，わかりやすく簡単に書きましたので，詳しい事を知りたい人は，以前本誌に連載された記事を見れば，かなり詳しくわかると思います．また，本誌創刊号からの合本も出版されていますので，そちらを参考にされても良いでしょう．

4 製作したキットの評価，雑感

写真 5 に私が製作したキットの完成写真を示します．このキットの第一の特長は，キーボードが付いていることだと思います．皆さんの中にもこのキーボードが手に入らず，困った方も多い事と思います．

また，キーボード部とコントロールパネル部が，うまく配置されており，デザイン的にも市販のシンセサイザに迫るものがあります．

ただ，自作マニアとしては，キットだけで満足していては腕が泣くというものです．このキットの回路部でも改良すれば，もっと良い結果が得られる所がいくつかあると思います．あとは自分で改良を加え，より効果的なシンセサイザを楽しんで下さい．

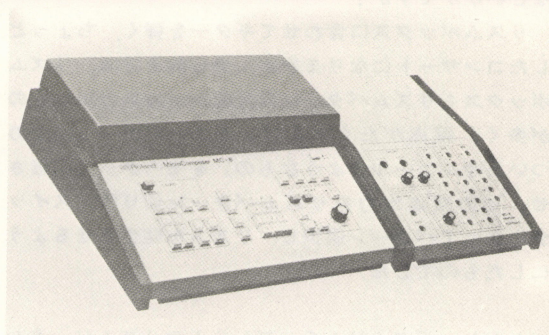
◘参考文献　小沢恭至他：「続電気楽器（音づくりの探究）」オーム社

チャッタレス・奥山のいいたいほうだい

今月のターゲット

マイコンを音楽に応用した製品の第1号機
ローランド MC-8

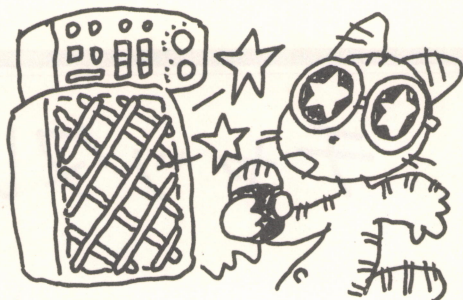
ローランドMC-8

ローランド社は，このほど8080プロセッサを用いた作曲器を発表した．これはアナログミュージックシンセサイザと組み合わせて使うもので，以前からあるシーケンサを発展させたものであるといえる．数値に置き替えた音階と音の長さ，そして音の大きさなどをキーによって入力し，それを電圧出力として取り出すものである．アナログシンセサイザには電圧の時間関数が複数個入力されるわけで，多様な音変化が望めることになる．

この種の製品の問題点は，従来からある音楽を作る場合いかに音楽の自然さを保つかということにあると思う．つまり良く言えば情緒あふれる人間味であり，悪く言えばぎこちなさとか気まぐれな感じをどれだけ演出できるか，である．その点MC-8は，音の長さを2種類（ひとつは音符と音符の間隔であり，もうひとつは音の鳴る時間である．）設定できることによりスタッカートからレガートまでの感じを良く出すことができる．また独自のフォーマットによるコーディング方法で，編集機能を持っている．そのため音を作りながら何度も手直しができるわけだ．

というわけで，マイコンの音楽へのアプリケーション第一号としては充分満足のいくもので，最近あちこちの週刊誌の誌面をにぎわしているエレクトーン自動演奏システムあたりとは雲泥の差である．まあアマとプロの違いとでもいうことになろうか．

ともかく，こうした企画を商品化する決断力は大いに評価されて良いといえる．ただし，ここでちょっと気になる点をあげると，お値断の120万円はソフト開発費を含めても少々高い様な気がするが，致し方ないのかもしれない．また，コンピュータの判断機能や演算機能を有効に生かしたプログラムも組みたいと思われるが，こんなことを思うのは少数のゲテモノマニア(?)くらいかな……

良くも悪くもマイコン専用機という面が強く出すぎていて，一部の人には不満を残すことになろうが，そういった人たちには自分でシステムを組んでもらうしかないだろう．ともあれ，開発し発売にふみきったという点だけでも大変に画期的なことであるといえる．

S8890を使って リズムボックスを作ろう

竹下　景

『シンセサイザによる音楽演奏も良いし，マイコンによる自動演奏も面白い，だけどもっと安価で手軽に電子音楽を楽しみたい』という方は，〝リズムボックス〟などいかがですか？

リズムボックスに合わせてギターを弾く．ちょっとしたコンサートになりますよ．少し前までは，リズムボックスもリズムパターン発生部がメカニカルなものが多く，接点がたくさん付いたスキャナー（接点のついたオルゴールのようなもの）をモーターで回転させ，その接点出力をリズムパターン切り替えスイッチによってパターン選択し，リズムを発生させるようにしたものでした．

しかし，近ごろは1チップＬＳＩのリズムジェネレータが出て来て，リズムボックスも作りやすくなっています．

今回はＳ8890を使用した，リズムボックス(注)を製作しました．図2はそのブロック図です．

Ｓ8890の出力は，各種のリズム楽器音の発生駆動信号を発生します．この信号はリズム選択スイッチで選択されたリズムで，テンポはＶＲで調整されたものです．音源回路は，Ｓ8890の出力信号により各種の打楽器音を発生します．図3はリズムパターンの一例です．

◆ リズムジェネレータＬＳＩ，S8890について

Ｓ8890はカウンター，デコーダーを内蔵したＲＯＭです．図4はＳ8890のブロック図です．

《S8890の特長》

❶　8種類のリズム楽器駆動信号を発生する．
❷　64のリズムパターンを発生する．
❸　10種類のリズムを選択する．
❹　7セグメントLED表示器でテンポの表示をする．

図1　旧タイプ・リズムボックスのブロック図

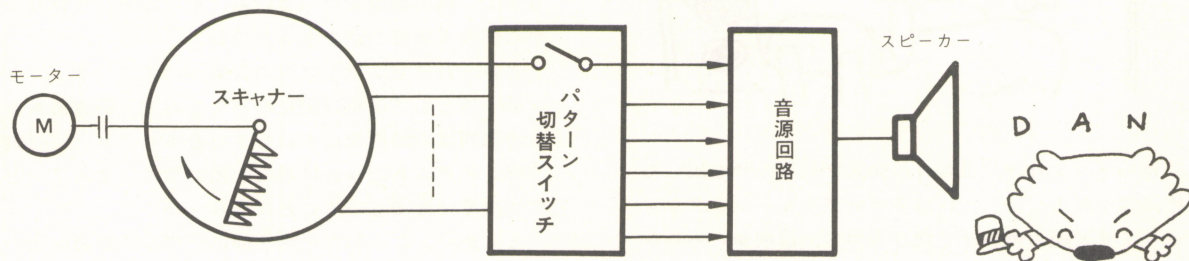



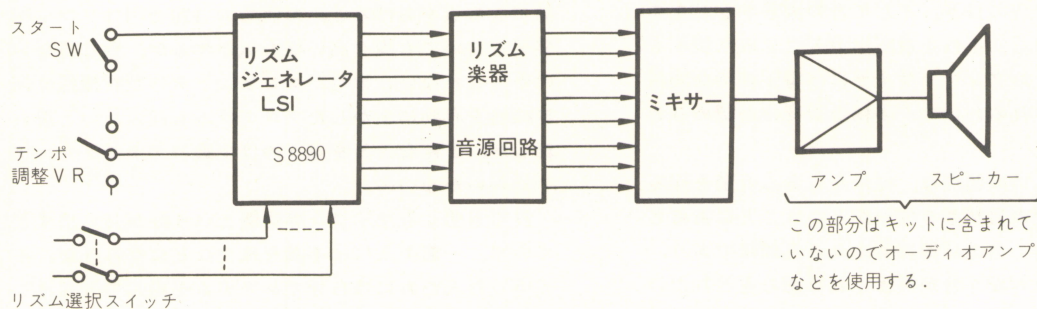
図2　リズムジェネレータキット（SK-302）のブロック図

(注)大阪の伸光(株)より販売されているリズムジェネレータ・キット（型番ＳＫ-302）を使用しました．

図3　リズムパターンの一例

図5　リズムとリズム楽器の種類

リズム	入力	出力	楽器
SAMBA	H1	F1	BASS DRUM
SWING	H2	F2	SNARE DRUM
SHAKE	H3	F3	FLOOR TAM
WALTZ	H4	F4	BONGO
JAZZ WALTZ	H5	F5	WOOD BLOCK
MARCH	H6	F6	CLAVES
JAZZ	H7	F7	CYMBAL
BOSSA NOVA	H8	F8	TAMBORINE
BEGINE	H9		
RHUMBA	H10		

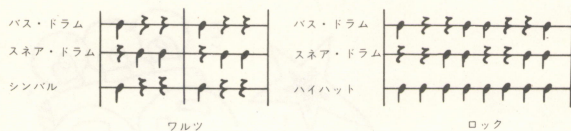



図4　S8890のブロック図

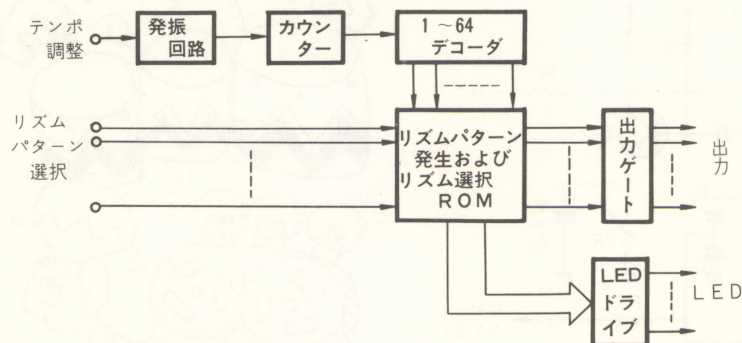
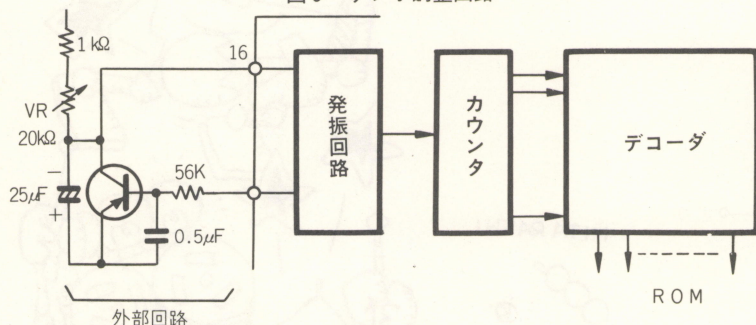



図6　テンポ調整回路

など，多くの特長を持っています．**図5**はリズムの種類とリズム楽器の種類です．

《テンポ調整》

テンポ調整は，外部に**図6**のような外部回路部を付けて，VR（20kΩ）により発振回路の発振周波数を調整します．カウンタは，6ビットのカウンタで，その出力を，1～64デコーダでタイミングに対応した信号に変換し，リズムパターン発生ROMに導かれます．

◘ 音源回路

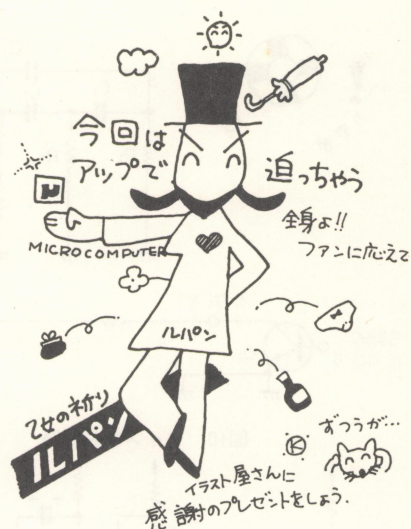



図7は各楽器の周波数を表にしたものです．音源回路にはいろいろありますが，要するに発振する回路であれば良いわけです．このキットに使用されている回路は二種類あります．一つは移相発振回路を利用した回路で，もう一つはノイズ発生器とフィルタを利用したものです．前者はバスドラム，フロアータム，ボンゴ，ウッドブロック，クラベスと，比較的にノイズの少ない音源回路に，後者はスネアードラム，シンバル，タンバリンなど，ノイズの多い音源回路に使用されています．

図8は移相発振回路を利用した音源回路です．

この発振回路は，〝VR〞を調整し，ゲインを1以下にし，通常は発振しないようにします．そしてS8890からの信号により，トランジスタQ1のベースにショックを与え，瞬間発振するようにしたものです．VRの抵抗値が大きくなるとゲインが下がります．またVRにより，エンベロープの調整もできます．発振周波数は，C_1とR_1で設定します．

図10はノイズ発生回路です．トランジスタのベースとエミッタを図のように逆に接ぎ，ノイズを発生させ，トランジスタアンプにて増幅する回路で，この出力に

図7　各楽器の周波数

楽　　器	周　波　数
BASS DRUM	60Hz
FLOOR TAM	130Hz
BONGO	190Hz
WOOD BLOCK	1,000Hz
CLAVES	2,300Hz
SNARE DRUM	3 ～5 kHz
CYMBAL	3 ～5 kHz
TAMBORINE	3 ～5 kHz

図8　移相発振回路を利用した音源回路

図10　ノイズ発生回路

出力

MITA RASHI

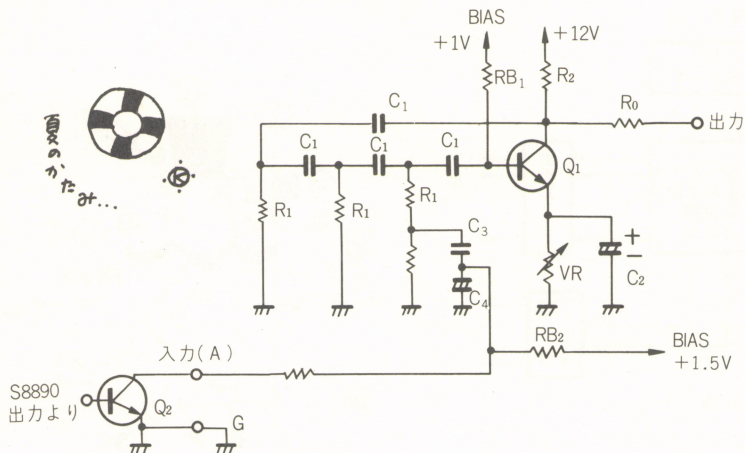
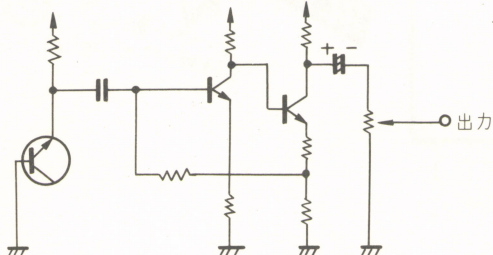
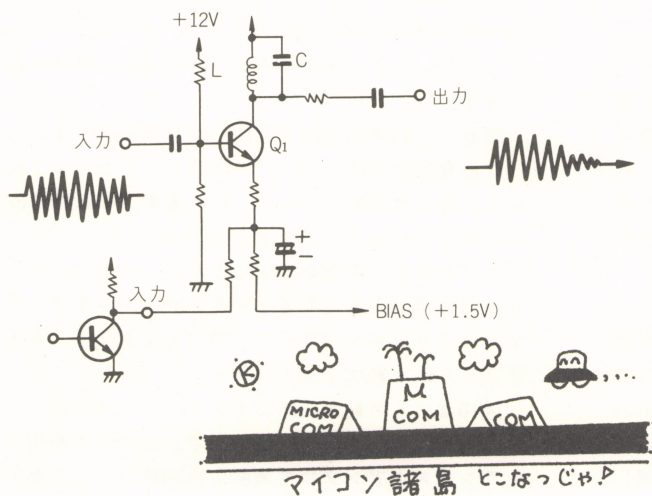



図11　バンドパス・フィルター

図9

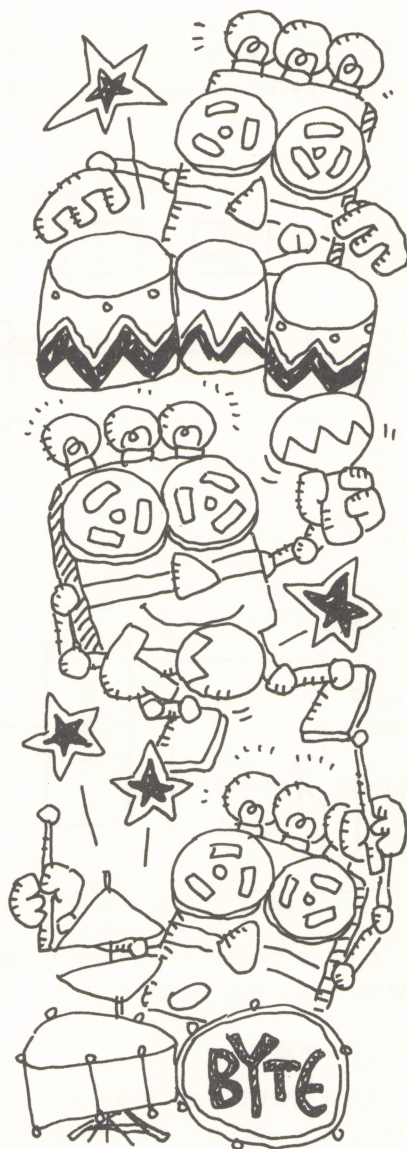
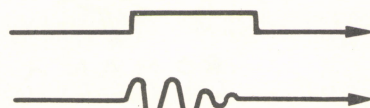
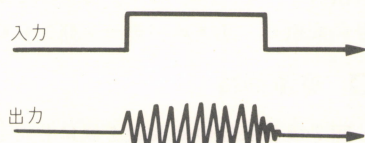
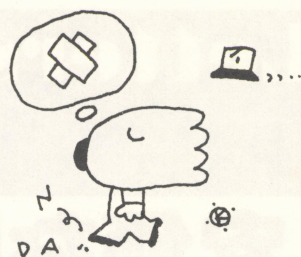
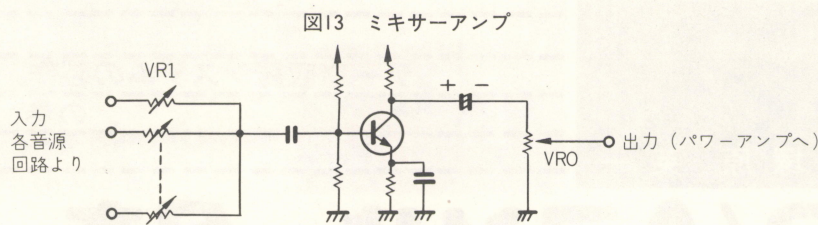
S8890の出力

音源回路の出力

図12

入力

出力

図13　ミキサーアンプ

は各種の周波数成分が含まれていますので，図11のバンドパス・フィルタを通し，必要な周波数成分を取り出します．周波数バンドはLとCで設定します．このバンドパス・フィルタもエミッタにバイアスを加え，通常は動作しないようにしてあり，S8890の出力信号により動作するようになっています．

◘ ミキサーアンプ回路

ミキサーアンプは各音源回路からの出力をミキシングし，増幅します．この出力はパワーアンプに接続し，音を出します．VR1は各音源回路の出力にはバラツキがありますので，これで調整します．

◘ 製作

このキットには，前記(1)～(4)までの各ブロックがすべて含まれています．製作は，PCボードにシルク印刷がしてあるので，マニュアルを参照して順次組み込めば問題ありません．調整は，自分の耳で音を聞きながら行ないます．マニュアル通り調整して行なえば特に問題ありませんが，この調整で音色の良し悪しが決まるので，あなたの腕の見せ所です．

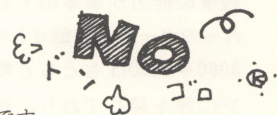
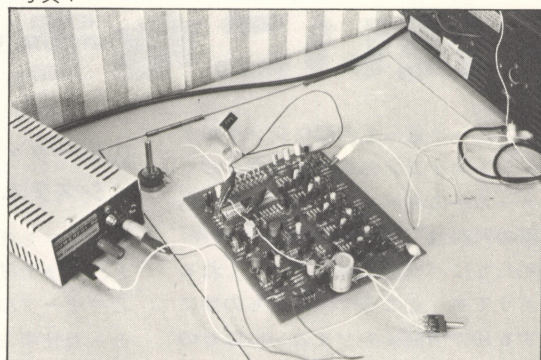



写真1

写真1に，組上がったリズム・ボックスのPCボード部を示します．あとは，適当なケースに組み込めばできあがりです．リズム切り替えSWは，ロータリーSWか，カーラジオに付いている押ボタンSWなどを付ければ良いでしょう．

NO　ドン　ゴロ

I/O バザール

〔売る〕
MC6871A　クロックジェネレーター用IC（IMHz）を1本のみ，￥4Kで．
〒154 東京都世田谷区太子堂3-33-3-321　木下清美　☎(03)414-6902

〔売る〕
TK-80＋TV32を￥110K～￥100Kで．
〒416 沼津市沼北町2-9-22
川合清正　☎(0559)21-7806

〔売る〕
新電源のSWレギュレータ　AY05004(5V, 4A)を￥16K．日置のデジタルテスタ3201を￥20K．3端子レギュレータ5V5Aタイプを￥3K（フェアチャイルド），W〒を待つ．
〒559 大阪市住之江区中加賀屋1-14-6　村田　洋

〔売る〕
プロセッサ・テクノロジ社SOL20（8080）を売ります（サイエンス誌の表紙にのったもの）．完成品．／TTY，LPT，ビデオディスプレイ用インターフェイス付．／キーボード，ビデオモニタ，磁気テープ付．／10KB RAMおよび2.5K ROMモニタ，OS付．／ソフトは機械語デモ用プログラム，BASIC付．／オプションとして追加メモリ，ディスクあり．
価格は上記のシステムで￥600K．
（J．マグアィアー）
買いたい人はI/O編集部へ☎ください．

〔売る〕
I/O創刊号～77年5月号までの7冊を￥2,800（〒共）で譲ります．
〒348 羽生市小須賀926　早川孝史

〔求む〕
アサヒダーストカラーアナライザー．三色分解フィルターがほしいので，他の部分はこわれていてもよい．￥10Kで，
〒235 横浜市磯子区中原4-17-3 西岡荘　坂倉幸彦

I/O 9

◘バザール投稿要領
官製ハガキに左のシールを貼り①売る，求む，交換の区別②品名③氏名④住所，〒を記入して下さい．

Letters

データ収録システムの製作の巻

塚原 英一

安価なD/Aコンバータを使いこなそう? 2

ここ一年足らずの間にシングルボード・マイコンの普及は目ざましく，メーカーのキャッチ・フレーズも"無限の可能性"，"創造性の追求"，等々，私たちにバラ色の夢を与えてくれるようです．ところで，私たちの手にする10万円前後のマイコンにはどの程度の能力があるのでしょうか．これらのボードに塔載されているCPU，8080や6800はもともと電卓やミニコンに源を発しており，かなり大容量のメモリとソフトウェアのサポートなしではパターン・デザインや競馬のシミュレーション予想のような創造的な仕事は困難と思われます．マイコンの現実的応用の主なものは電子楽器のプログラム演奏に代表されるシーケンス・コントローラ，CRTなどを利用したディスプレイ，データ・ファイル，タイマー的応用，そして本稿で取り上げる計測への応用です．

◆データ・アクイジション・システム

マイコンを記録計として利用すると，今まで私たちが時計とにらめっこでデータ取りをしなければならなかった仕事を正確に処理してくれます．

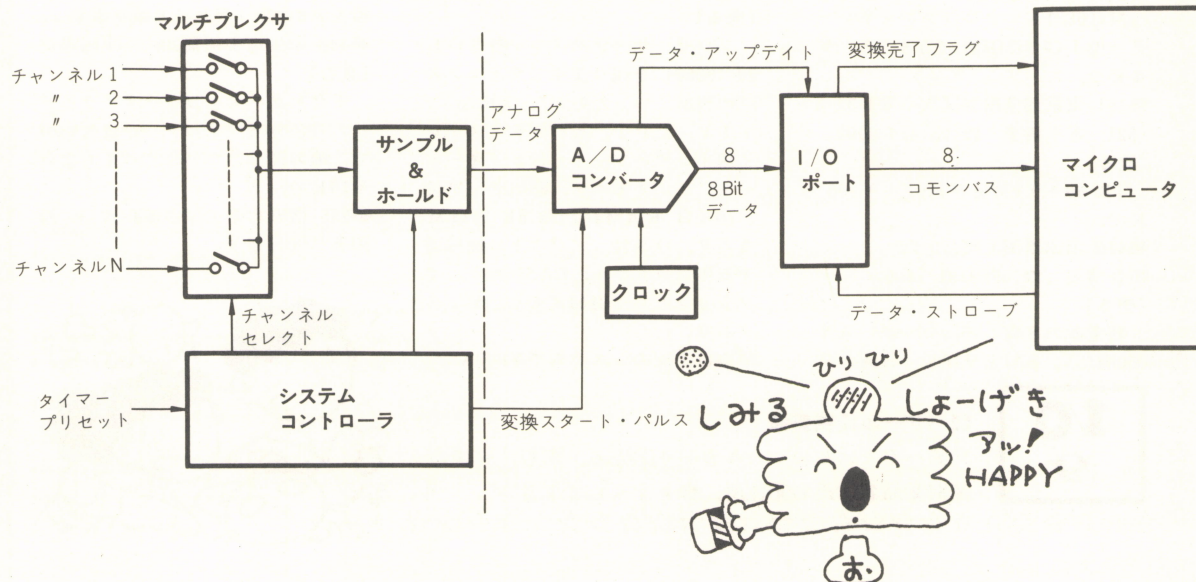
余った時間で無限の可能性を追求するか，コーヒーを飲みに行くかは意見の別れるところでしょう．

図1はデータ・アクイジション・システムのブロック図です．まずアナログ入力のチャンネル数，アクイジション・インターバルなどによって点線から左の部分の仕様が決りますが，この部分は次回のテーマにさせていただくこととして，本稿では点線から右側の部分，つまりA/Dコンバータおよびマイコンとのインターフェイスを取り上げてみました．今回製作したA/Dコンバータのコンバージョン・タイムは5 μs程度ですから，アナログ入力が数μs以内で急変しない事を条件にした1チャンネル・データ・アクイジション・システムです．

◆遂次比較型A/Dコンバータ

図2に遂時比較型A/Dコンバータのブロック図を，図3にタイミング・チャートを示します．D/Aコンバータの動作については先月号で御紹介したとおりなので省略させていただくことにして，図をみながら，A/Dコンバージョンの行なわれる過程を追ってみましょう．

まずスタートパルスの"LOW"期間中最初のクロックの立上り①によって$\overline{cc}$及びレジスタQ_0～Q_6が"High"に，Q_7が"LOW"にセットされます．こ

図1　データ・アクイジション・システム

れがA/Dコンバージョン・ループのリセットされた状態です．このときD/Aコンバータの出力は，フルスケール＋10Vとして，10〔V〕×127/256＝4.96〔V〕になっています．そこでコンパレータはアナログ入力とD/Aコンバータの出力を比較し，アナログ入力が大きいときはコンパレータの出力を“High”に，逆のときは“LOW”にセットします．さて，次のクロックの立上り②によってレジスタの“D”に入力されたコンパレータの出力が取り込まれてQ_7の出力として固定されます．同時にQ_6が“LOW”になり，Q_0～Q_5は“High”のままで次のクロックの立上り③を待ちます．③のタイミングでも同様な方法でQ_6の出力が固定され，次の下位ビット（ここではQ_5）が“LOW”になります．まったく同様な操作がタイミング⑨によってQ_0の出力が固定されるまで続けられますが，⑨ではその他に変換終了フラグ$\overline{cc}$を“LOW”に落します．もうご理解されたことでしょうが，このA/DコンバータはD/Aコンバータの出力とアナログ入力とを比較しながらバイナリー・デジタル・コードのMSB（最上位ビット）から順に決定してゆく方法で，比較的高速のA/Dコンバータとして広く採用されている方式です．余談ですが，AMD社の逐次比較レジスタは多くのA/Dモジュール・メーカーで標準的に使用されています．

◆実験回路

図1のブロック図の点線から右の部分を実現したのが図4に示した実験回路です．D/A部分は先月号で御紹介した回路をそのまま流用していますが，出力I_0がコンパレータLM311の入力部でダイオード・クランプされていますのでフルスケール調整は$\overline{I_0}$をモニタしながら行います．つまり，Am2502をICソケットから抜くとAmDAC-08の入力はすべて“High”になりますから，VinをGND（0V）

図2　逐次比較型ADコンバータ

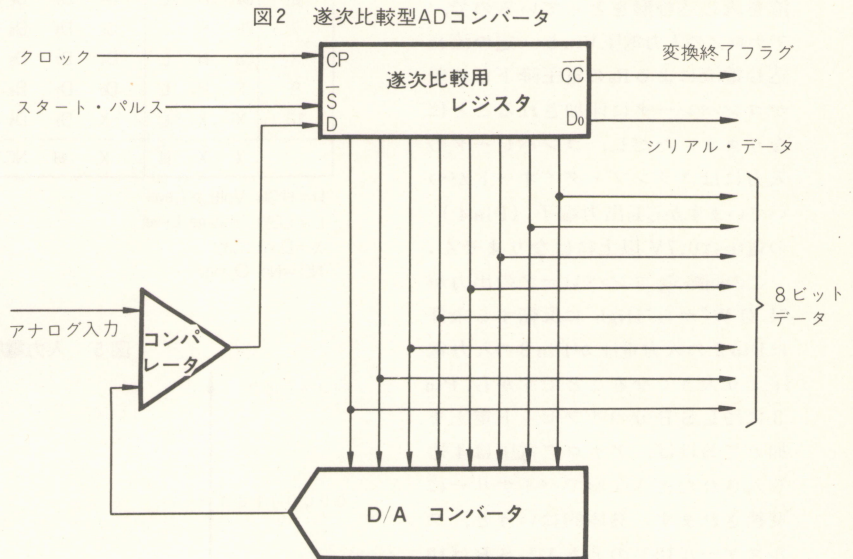
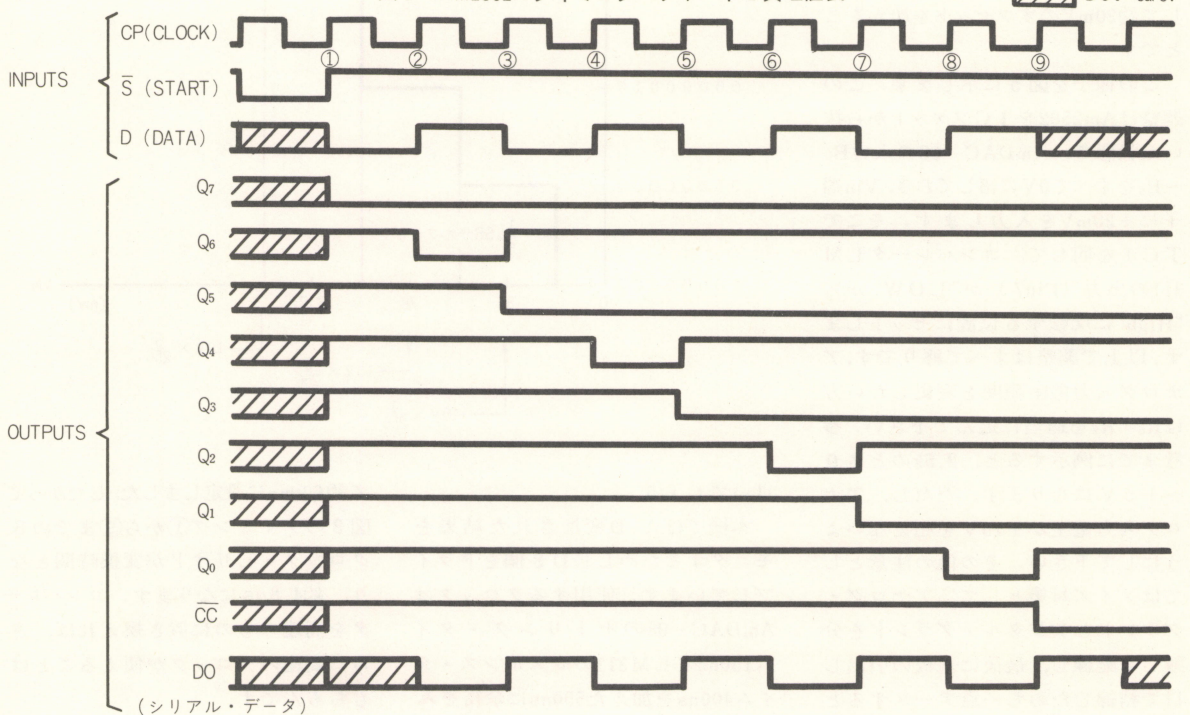



図3　Am2502のタイミング・チャートと真理値表

に落した状態でＴＣ２を回して，$\overline{I_0}$出力端子の電圧を０Ｖに追い込めば良いわけです．フルスケール調整が済みましたら次はコンパレータのオフセットを調整します．本機の入力電圧範囲は０～＋10Ｖですが，この入力電圧がR_6，R_7を通してＤ/Ａコンバータの出力に接続されています．

Ｄ/Ａコンバータの出力は与えられたデジタル・コードに相当する一定電流を流し込む形をとっているので，アナログの入力電圧Vinと一定の流れ込む電流によるR_6の電圧降下との差がコンパレータに印加されることになります．ただし，コンパレータの入力にはクランプ・ダイオードがついていますからI_0出力端子（Pin４）の電圧は0.7V 以上にはなりません．

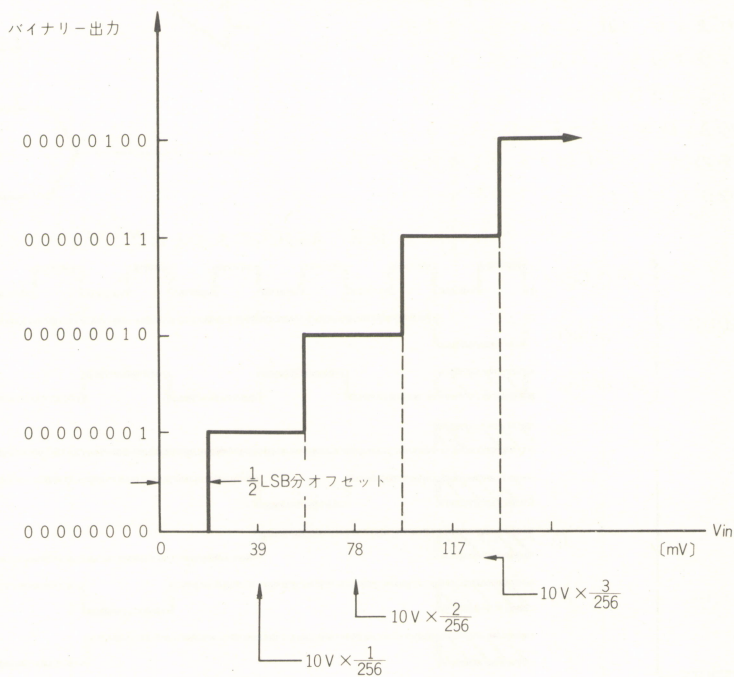
この回路で，コンパレータの出力が“ＬＯＷ”から“High”に反転する条件はPin２の入力電圧がPin３の入力電圧より大きくなることですから，Pin３に½ＬＳＢ分のオフセット電圧を加えておけば，アナログ電圧は４捨５入されたような形でバイナリーに変換されます．具体的にいうと，フルスケール10Ｖのとき１ＬＳＢは10Ｖ÷256≒39mＶですから½ＬＳＢとして約20mＶのオフセットを加えることです．

この様子を図５に示します．この調整はAm2502をＩＣソケットから抜いた状態で，AmDAC－08の入力B_1～B_8をすべて0Ｖに落しておき，Vin端子に＋20mＶを入力します．そこでＴＣ１を回して，コンパレータＬＭ311の出力（Pin7）が“ＬＯＷ”から“High”に反転する位置にセットします．以上で調整はすべて終りです．アナログ入力電圧範囲を変更したい方は$R_6=R_7$を適当に選んで下さい．参考までに例示すると，2.5kのとき０～＋５Ｖになります．ただし，アナログ入力電圧が±15Ｖを超えないようにして下さい．その他の注意としてはノイズ対策として，アナログ・グランドとデジタル・グランドを分離して配線し，最後に基板の引出し口で結線したのち一点アースすると

Am2502真理値表

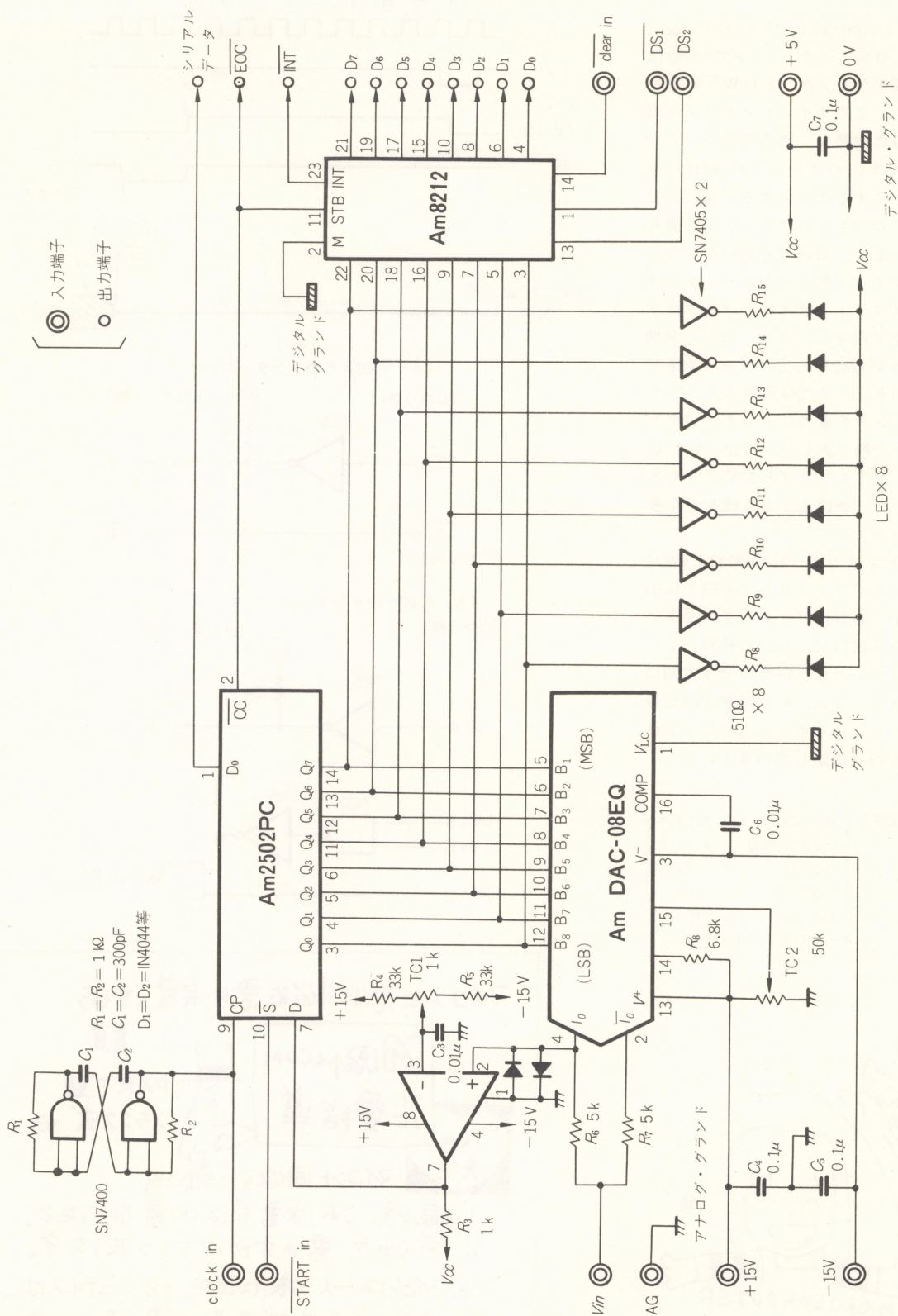
時刻	入力			出力									
tn	D	S	E	D_0	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	CC
0	X	L	L	X	X	X	X	X	X	X	X	X	X
1	D_7	H	L	X	L	H	H	H	H	H	H	H	H
2	D_6	H	L	D_7	D_7	L	H	H	H	H	H	H	H
3	D_5	H	L	D_6	D_7	D_6	L	H	H	H	H	H	H
4	D_4	H	L	D_5	D_7	D_6	D_5	L	H	H	H	H	H
5	D_3	H	L	D_4	D_7	D_6	D_5	D_4	L	H	H	H	H
6	D_2	H	L	D_3	D_7	D_6	D_5	D_4	D_3	L	H	H	H
7	D_1	H	L	D_2	D_7	D_6	D_5	D_4	D_3	D_2	L	H	H
8	D_0	H	L	D_1	D_7	D_6	D_5	D_4	D_3	D_2	D_1	L	H
9	X	H	L	D_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	L
10	X	X	L	X	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	L
	X	X	H	X	H	NC	NC	NC	NC	NC	NC	NC	NC

H＝HIGH Voltage Level
L＝LOW Voltage Level
X＝Don't Care
NC＝No Change

図５ 入力電圧VSデジタル出力

良いでしょう．

本機ではＡ/Ｄ変換された結果をモニタするためＬＥＤ８個をドライブしています．使用するクロックはAmDAC－08のセトリング・タイム150nsとＬＭ311のレスポンス・タイム400nsを加えた550nsに余裕をみて約600nsに設定しました．したがって図３のタイミング①から⑨までの８クロック・ピリオドが変換時間となり，約4.8μsになります．コンパレータを高速のものに置き換えれば，さらに高速のクロックが使えることはもちろんです．

図4　実験回路

Am8212
Am2502PC
Am DAC-08EQ
シリアルデータ
EOC
INT
clear in
DS1
DS2
+5V
0V
入力端子
出力端子
デジタル・グランド
アナログ・グランド
SN7405×2
LED×8
51Ω×8
SN7400
R1＝R2＝1kΩ
C1＝C2＝300pF
D1＝D2＝IN4044等
clock in
START in
Vin
AG
+15V
−15V
Vcc

◆マイコン・インターフェイス

さて，変換が終了すると$\overline{\mathrm{cc}}$の“High”→“ＬＯＷ”のタイミングでAm8212の$\overline{\mathrm{INT}}$ラインが“ＬＯＷ”に落されます．このラインをマイコンの割込入力に接続し，割込処理ルーチン内でこのＩ/Ｏポート（Am8212）をアクセスするようにすると良いでしょう．このときのタイミングを図６に示します．割込みを使いたくない方はマイコン側に出力ポートを設け，変換スタート・タイミングをプログラム上で指定し，クロックから変換終了時刻を計算してアクセスするソフト的方法，または変換スタート・タイミングを適当なインターバル・タイマで操り返し与えながらＣＰＵ側は$\overline{\mathrm{INT}}$をフラグとしてアクセス・タイミングを決定する方法などが考えられます．

最後に，出力ポートAm8212とCPUのインターフェイス方法を図７に示します．必要なゲートはクロックに使ったＳＮ7400，及びＬＥＤのドライブに使ったＳＮ7405の余りで間に合います．なお，ＣＰＵ側の$\overline{\mathrm{cs}}$信号はアドレス・ラインからデコードされたものを想定しています．

〔ＩＣ価格〕

Am2502PC	￥ 4,500＊
AmDAC-08EQ	￥ 4,300＊
LM311H	￥ 700
Am8212PC	￥ 1,200

サイエンス・システム・サポート社
＊一般の店にはありません．

図６　出力ポートのアクセス

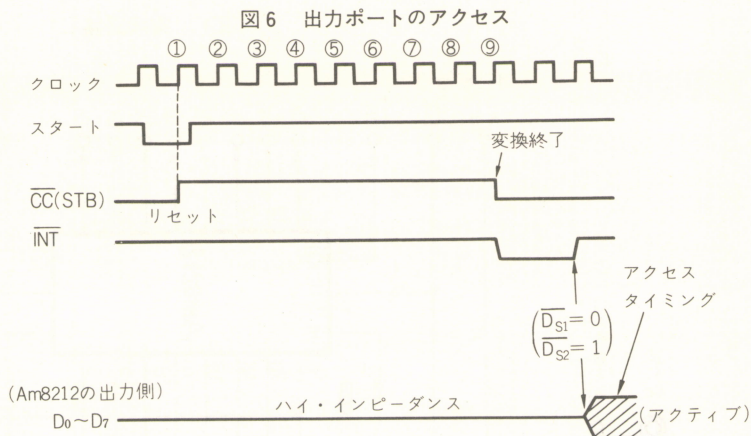
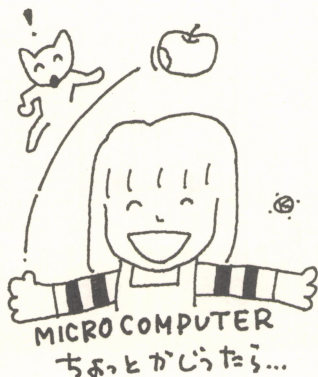
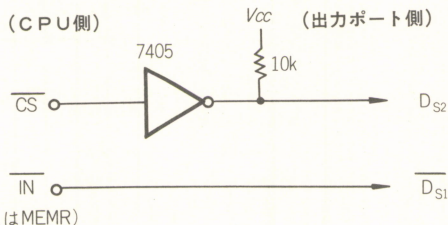
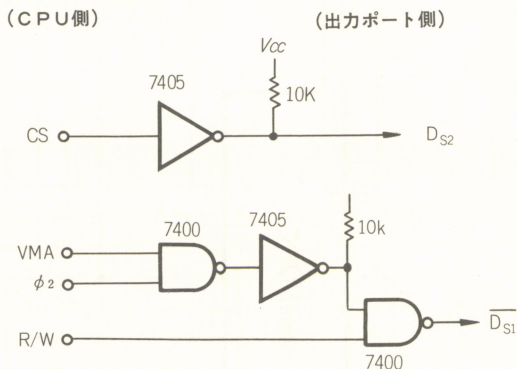
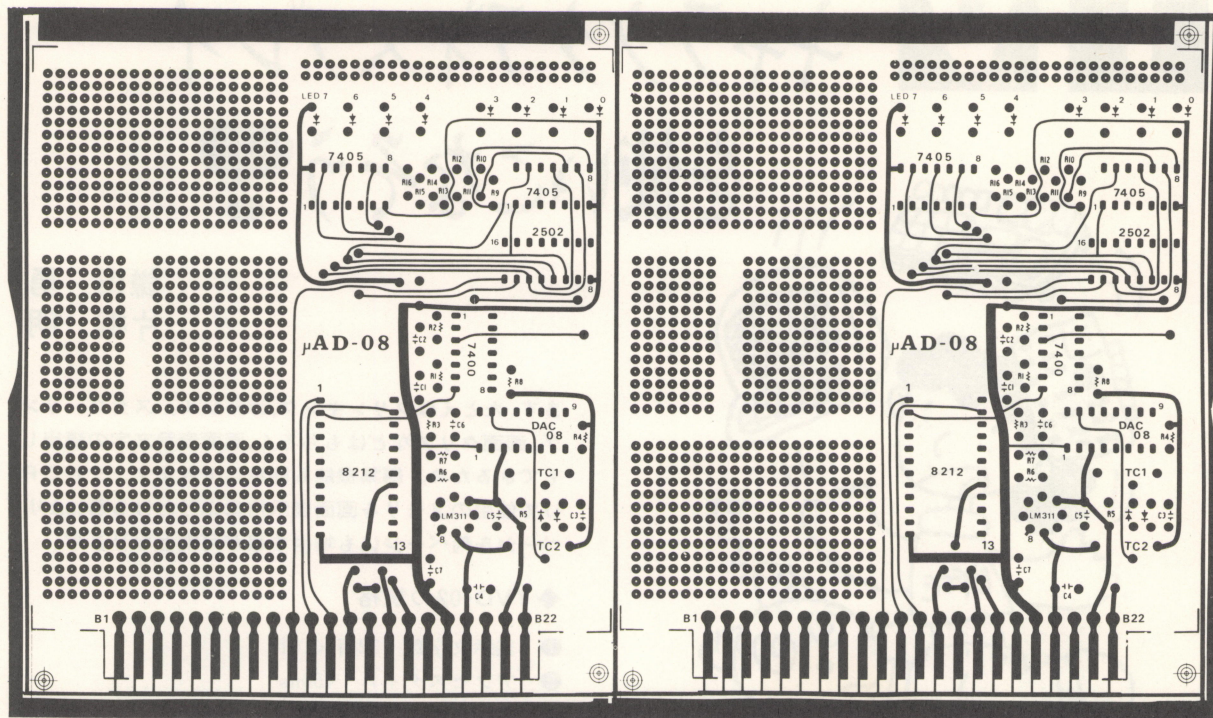
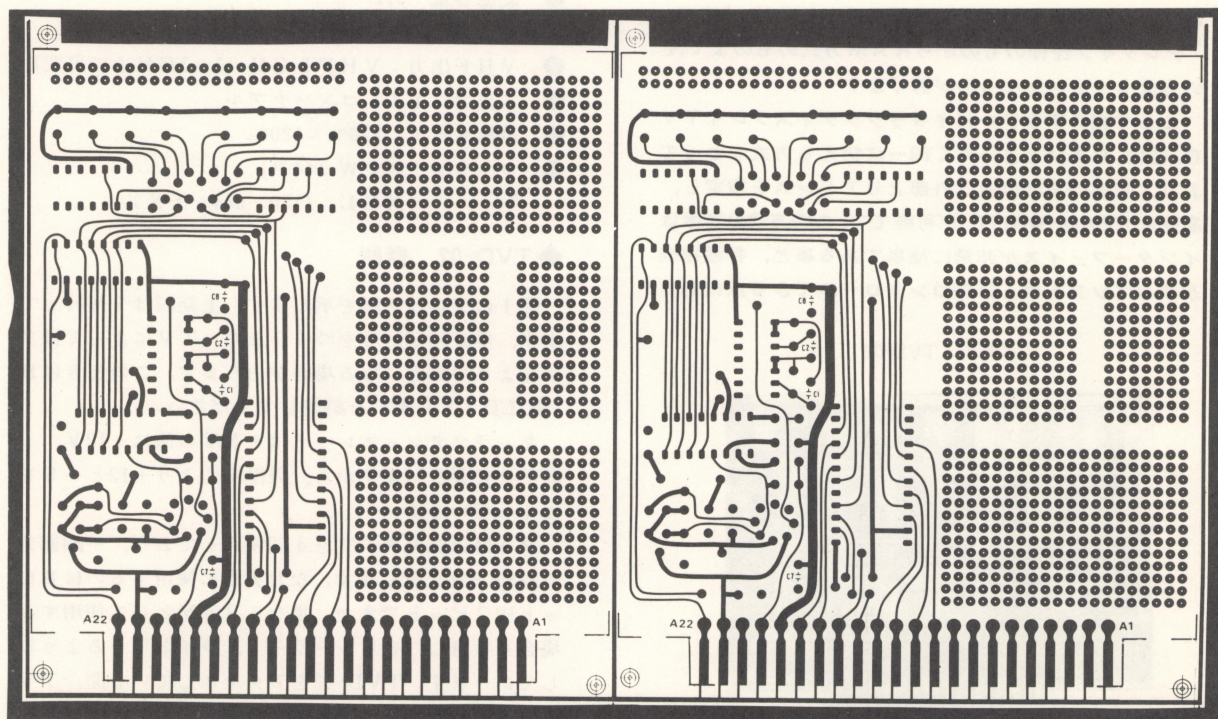



図７
（Ａ）8080Ａとのインターフェイス

（Ｂ）6800とのインターフェイス

プリントパターン（表）

プリントパターン（裏）

TV キャラクタディスプレイを使いこなそう❢

鎌田　勇
片桐　明

最近，キャラクタディスプレイ装置がかなり出まわってきて，アマチュアでも手軽にマイコンのI/Oとして使用できるようになってきました．種類はいろいろあり，ブラウン管つきのものから単なるユニットもの，テレタイプ仕様のものからRAM方式のものまで，その方式もさまざまなものがあります．

ここで紹介するテレビキャラクタディスプレイTVD-02は，RAMとまったく同一に扱える方式をとっており，画面の文字位置は外部よりアドレスを指定し，高速なREAD，WRITEが可能です．この方式の特徴はインターフェイスが非常に簡単である事と，各種機能がすべてソフトウェアでコントロールできる点にあります．たとえばブリンキング，カーソル，スクローリング，画面クリアなどはもちろん，画面表示文字の読出しもできるため，編集機能も簡単にできます．またCPUで使用のメモリを画面表示用にふりわける事により，ページを何ページにも増やす事が可能となります．

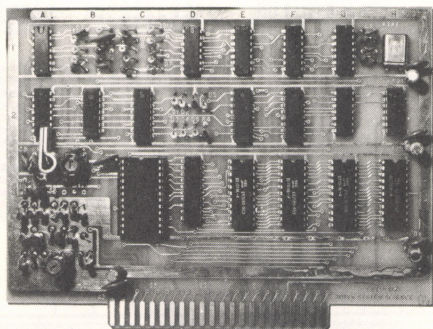



写真１　TVD-02

◆TVD-02の規格

❶ 表示文字数：32桁×16行
❷ アクセスタイム：450ns
❸ 表示文字の種類：英数字，特殊記号，カタカナ全128種（オプションで英小文字，ギリシャ文字なども可）
❹ ドットサイズ：７×９ドット
❺ 記憶容量：512バイト
❻ 電　源：＋５Ｖ　単一電源　0.6A
❼ ＶＨＦ出力：ＶＨＦ２ＣＨ（1～3ＣＨまで可変）
❽ 入出力：ＴＴＬコンパチブル
❾ 基板サイズ：170㎜×120㎜
❿ コネクタ：22ＰＷ（44Ｐ）
　　ＫＥＬ　1150－044－ＸＸＸ

◆TVD-02　概説

図１にブロック図を示します．主発振は７ＭＨｚで水平，垂直同期信号をつくります．ＴＶによっては多少左よりの画面になる場合があります．このときはH-HOLD（ＴＶ側）を調整してください．

キャラクタジェネレータは＋５Ｖの他に＋12Ｖ，－５Ｖを必要としますから，発振出力より＋12と－５をつくっています．

ＲＡＭは2101タイプを４個使用しており，一画面分512バイトを記憶します．なお実際に使用するのは８ビット中７ビットですが，単なるＲＡＭとして使用する場合も考慮し，８ビットリード，ライトできるようにしてあります．TVD-02をアクセスしている間，ＴＶ画面は指定キャラクタを表示できませんが，アクセスの時間はμsのオーダであり，通常の使用ではまったく

図1　TVD-02　ブロックダイヤグラム

図2　TVD-02　コネクタ接続図

DB_0～DB_7	：データバス(入、出力)
AB_0～AB_8	：アドレスバス(入力)
$\overline{CS}$	：チップセレクト(入力)
$\overline{WR}$	：ライトストローブ(入力)
OD	：アウトプットディゼーブル(入力)
Vcc	：＋5V　0.5A
GND	：GND

図3　電気的特性

気になりません．同期信号と合成された映像信号はＶＨＦモジュレータによりＴＶのアンテナ端子に供給されます．一応２ＣＨに調整していますが１～３ＣＨまで可変です．

◆TK-80との具体的接続方法

図５にＫＢ-02Ｌと組み合せ，ＴＫ-80でＴＶタイプライタを構成した場合の回路図を示します．キーボードはポートＢとポートＣを使用します．ＴＶＤ-02の上位アドレスは74ＬＳ30でデコードし，ＦＥ00～FFFFのアドレスをアクセスできるようにします．この場合ＴＫ-80を若干改造しなくてはなりません．

図でNo.９のインバータはＴＫ-80内の74ＬＳ04で,6個のインバータのうち１個分余っているのでそれを使用し，デコーダ2155の14ピンに接続します．No.5のμＰＢ215はＤＭＡのときＴＫ-80のＲＡＭを選択するためのものです．その他使用上の注意としては，ＴＶのアンテナ端子に接続するとき，トランスレスＴＶの場合を考慮して，50ＰＦ程度のコンデンサを各々直列に入れます．

◆TVD-02のソフトウェア

ＴＶＤ-02，ＫＢ-02Ｌと組み合わせて，**図５**のようなターミナル・システムを構成した場合のソフトウェアを紹介します．

(1)　キー入力サブルーチン
(2)　ＴＶ出力サブルーチン
(3)　カーソル・メンテナンスサブルーチン
(4)　クリアサブルーチン
(5)　スクロールサブルーチン

図6　キャラクタ・コードフォーマット

❶　キー入力サブルーチン

入力データ：なし

出力データ：レジスタＡに，キャラクタコード（7bit ASCII＋カナ bit）

CALL IN33（ＣＤ3903）

レジスタ　ＢＣ，ＤＥ，ＨＬ　は保存されます．

このサブルーチンは，コールされるとまず，カーソルポインタで示す画面位置に，カーソル“－”を表示し，それを点滅します．

キーボードのキーが押されると，カーソルを元のキャラクタに戻した後，レジスタＡにキャラクタコード（様式は別記）をセットし，ユーザにリターンします．

“CTRL” KEY ＋ “C” KEY を押した場合は，画面クリアのみを行い，ユーザへはリターンしません．

❷　ＴＶ出力サブルーチン

入力データ：レジスタＡに，キャラクタコード（7bit　ASCII＋カナbit）

出力データ：なし

CALL　OUT33（ＣＤ0003）

レジスタ　ＢＣ，ＤＥ，ＨＬ　は保存されます．

このサブルーチンは，コールされると，レジスタＡの内容を，キャラクタコードとみなし，対応する文字をＴＶ画面に表示します．

文字が表示される画面位置は，カーソルポインタの内容により決定されます．文字表示後，カーソルポインタはインクリメントされます．

次のコードに対しては，特殊な処理を行います．

(イ)　キャリッジ・リターン（ＣＲ）コード“０Ｄ”
　　カーソルを左端に移す

(ロ)　ラインフィード（ＬＦ）コード“０Ａ”
　　カーソルを１行下に移す

❸　カーソル・メンテナンス・サブルーチン

入力データ：レジスタＡにカーソル移動量　出力データ：キャリーフラグ＝1　ノーマル＝0　カーソルが，画面サイズを越えた場合CALL CSMOD（CD8B03）

レジスタ　ＢＣ，ＤＥ，ＨＬ　は保存されます．

このサブルーチンは，レジスタＡの内容を，2つの補数型式の符号＋7bit のバイナリーデータと解釈し，カーソルポインタの値を，レジスタＡの値で増減します．

〔例〕

レジスタＡ	ＣＡＬＬ前のポインタ	ＣＡＬＬ後のポインタ
12（＋18(10)）	01Ａ７	01Ｂ９
Ｅ０（－32(10)）	01Ａ７	0187

（カーソルが１行上ります）

カーソルポインタの有効値は，0000～01ＦＦです．0000が，画面左上隅に対応し，01ＦＦが画面右下隅に対応します．

このサブルーチンは，ユーザが，カーソルを任意の画面位置に移動させる時，使用します．

なお，カーソルポインタ変更の結果が，0200～021Ｆになった時は，自動的に更に“Ｅ0”が，加えられ，01Ｅ0～01ＦＦに修正されます．

この時，キャリーフラグがリセットされて，リターンします．

❹　クリアサブルーチン

入力データ：なし

写真5　CRTデバッガー

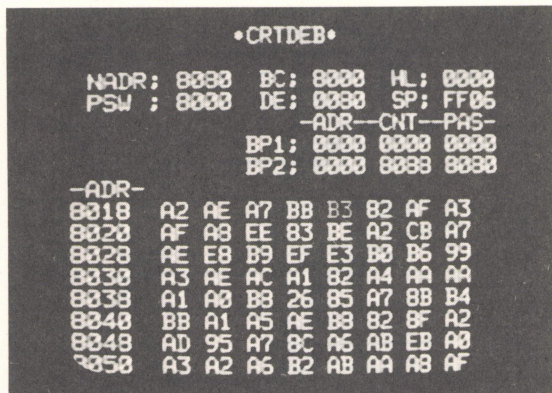
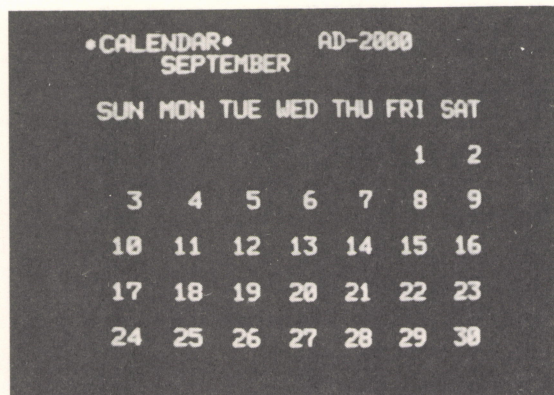



写真6　万年カレンダー西暦2000年の9月

出力データ：なし

CALL CLR33 (CDB803)

レジスタ　BC, DE, HL　は保存されます.

❺ スクロールサブルーチン

入力データ：なし

出力データ：なし

CALL SCROL (CDD903) レジスタはすべて破壊されます.

このサブルーチンは，画面のすべてを一行ずつ上方にくり上げ，最下行にブランクを挿入します.

注)『OUT33』において，キャラクタを表示した位置が，最終位置（画面右下隅）の場合,「OUT33」の内部で，自動的に「SCROL」が，CALLされます.

◆データ・フォーマット

「IN33」,「OUT33」で使用するキャラクタ・コードは，図6に示す形式です.

PROM-04　ターミナルサービスプログラム

```
ADDRESS LABEL  OPERATION OPERAND  DEC   OCTAL   COMMENT
        ;      PRŌM      04
        ;      TERMI     NAL      SER   VICE    PRŌGRAM
        ;
0300    ŌUT33  PUSH      B              C5
   1           PUSH      D              D5
   2           PUSH      H              E5
   3           CPI       0D             FE0D    ;CHECK "CR"
   5           JNZ       ŌU10           C2120 3
        ;
   8           LXI       H, PNTR        21A08 3
   B           MŌV       A, M           7E
   C           ANI       E0             E6E0
   E           MŌV       M, A           77
   F           JMP       LŌAD           C3B40 3
        ;
0312    ŌU10   CPI       0A             FE0A
   4           JNZ       ŌU20           C21C0 3
   7           MVI       A, 20          3E20
   9           JMP       ŌU50           C32D0 3
        ;
031C    ŌU20   CALL      PNTSET         CDD10 3
   F           MVI       B, 3F          063F
0321           ANA       A              A7
   2           JP        ŌU30           F2290 3
   5           ANA       B              A0
   6           ŌRI       40             F640
   8           MŌV       B, A           06
        ;
   9    ŌU30   ANA       B              A0
   A           MŌV       M, A           77
   B           MVI       A, 01          3E01
        ;
   D    ŌU50   CALL      CSMŌD          CD8B0 3
0330           JC        ŌU60           DA360 3
   3           CALL      SCRŌL          CDD90 3
        ;
   6    ŌU60   JMP       LŌAD           C3B40 3
0339    IN33   PUSH      B              C5
   A           PUSH      D              D5
   B           PUSH      H              E5
   C    IN10   CALL      PNTSET         CDD10 3
   F           MŌV       A, M           7E
0340           MVI       M, 20          3620
   2           PUSH      PSW            F5
   3           MVI       B, 01          0601
   5           MŌV       D, B           50
   6    IN11   CALL      SCAN           CD790 3
   9           JZ        IN11           CA460 3
   C    IN12   CALL      SCAN           CD790 3
   F           JNZ       IN12           C24C0 3
        ;
0352           PŌP       PSW            F1
   3           MŌV       M, A           77
        ;
   4           IN        01             DB01    ;INPUT PŌRT B
   6           CPI       03             FE03    ;CHECK CLEAR"
   8           JNZ       IN13           C2610 3
035B           CALL      CLR33          CDB80 3
   E           JMP       IN10           C33C0 3
        ;
0361    IN13   MŌV       B, A           47
   2           XRI       80             EE80
   4           CPI       E0             FEE0
   6           JC        IN14           DA6C0 3
   9           ANI       5F             E65F
   B           MŌV       B, A           47
        ;
   C    IN14   MŌV       A, B           78
   D           ANI       7F             E67F
   F           MŌV       B, A           47
0370           MŌV       A, E           7B
   1           RRC                      0F
   2           RRC                      0F
   3           ANI       80             E680
   5           ŌRA       B              B0
   6           JMP       LŌAD           C3B40 3
        ;
```

```
ADDRESS LABEL  OPERATION OPERAND  DES   OCTAL   COMMENT
        ;
0379    SCAN   DCX       B              0B
   A           MŌV       A, B           78
   B           ŌRA       C              BI
   C           JNZ       SC30           C2860 3

   F           MŌV       A, M           7E
0380           ADD       D              82
   1           XRA       D              AA
   2           SUB       D              92
   3           MŌV       M, A           77
   4           MVI       B, 20          0620
        ;
   6    SC30   IN        02             DB02
   8           MŌV       E, A           5F
   9           ANA       D              A2
   A           RET                      C9
038B    CSMŌD  PUSH      B              C5
   C           PUSH      D              D5
   D           PUSH      H              E5
   E           LXI       H, PNTR        21A08 3
0391           ANA       A              A7
   2           JM        CS20           FA9F0 3

   5           ADD       M              86
   6           MŌV       M, A           77
   7           INX       H              23
   8           MŌV       A, M           7E
   9           ACI       00             CE00
   B           MŌV       M, A           77
   C           JMP       CS30           C3A70 3
        ;
   F    CS20   ADD       M              86
03A0           MŌV       M, A           77
   1           INX       H              23
   2           CMC                      3F
   3           MŌV       A, M           7E
03A4           SBI       00             DE00
   6           MŌV       M, A           77
        ;
   7    CS30   CPI       02             FE02
   9           JNZ       LŌAD           C2B40 3
   C           MVI       M, 01          3601
   E           DCX       H              2B
   F           MŌV       A, M           7E
03B0           ŌRI       E0             F6E0
   2           MŌV       M, A           77
   3           ANA       A              A7
        ;
   4    LŌAD   PŌP       H              E1
   5           PŌP       D              D1
   6           PŌP       B              C1
   7           RET                      C9
        ;
03B8    CLR33  PUSH      B              C5
   9           PUSH      D              D5
   A           PUSH      H              E5
   B           LXI       D, 0200        11000 2
   E           LXI       H, VRAM        2100F E
        ;
03C1    CL10   MVI       M, 20          3620
   3           INX       H              23
   4           DCX       D              1B
   5           MŌV       A, E           7B
   6           ŌRA       D              B2
   7           JNZ       CL10           C2C10 3
   A           XCHG                     EB
   B           SHLD      PNTR           22A08 3
   E           JMP       LŌAD           C3B40 3
        ;
03D1    PNTSE  T LHL     D, PNTR        2AA08 3
   4           LXI       D, VRAM        1100F E
   7           DAD       D              19
   8           RET                      C9
        ;
03D9    SCRŌL  LXI       D, 01E   0     11E00 1
   C           LXI       H, VRA   M     2100F E
   F           LXI       B, VRA   M+2   0 012 0FE
        ;
03E2    LI10   LDAX      B              0A
   3           MŌV       M, A           77
   4           INX       B              03
   5           INX       H              23
   6           DCX       D              1B
   7           MŌV       A, E           7B
   8           ŌRA       D              B2
   9           JNZ       LI10           C2E20 3
        ;
   C           MVI       B, 20          0620
   E    LI12   MVI       M, 20          3620
03F0           INX       H              23
   1           DCR       B              05
   2           JNZ       LI12           C2EE0 3
   5           RET                      C9
```

ピッピ

MICRO
COMPUTER

CALENDAR(FOR TVD-02&KB-02L)

```
START  8000 MVI  A,93H     3E 93
          2 OVT  03H       D3 03
RSTRT  8004 CALL CLR33     CD B8 03
          1 LXI  H,DATA1   21 A0 81
          A CALL PRINT     CD 82 81
INIT   800D MVI  B,30H     06 30
          F MOV  C,B       48
         10 MOV  D,B       50
          1 MOV  E,B       58
LOOP1  8012 CALL IN33      CD 39 03
          5 MOV  H,A       67
          6 ANI  F0H       E6 F0
          8 CPI  30H       FE 30
          A JNZ  NEXT1     C2 24 80
          D MOV  B,C       41
          E MOV  C,D       4A
          F MOV  D,E       53
         20 MOV  E,H       5C
          1 JMP  LOOP1     C3 12 80
NEXT1  8024 MOV  A,H       7C
          5 CPI  20H       FE 20
          7 JNZ  ERROR     C2 59 81
          A LXI  H,DATA2   21 B0 81
          D CALL PRINT     CD 82 81
         30 MOV  A,B       78
          1 CALL CHNG      CD 8D 81
          4 CALL B         CD 98 81
          7 MOV  B,A       47
          8 MOV  A,D       79
          9 CALL CHNG      CD 8D 81
          C ADD  B         80
          D MOV  D,A       67
          E MOV  A,C       7A
          F CALL CHNG      CD 8D 81
         42 CALL LSHFT     CD 98 81
          5 MOV  DA        57
          6 MOV  A,E       7E
          7 CALL CHNG      CD 8D 81
          A ADD  D         82
          B MOV  L,A       6F
          C XCHG           EB
Y ADJ  804D MVI  B 05H     06 05
          F MVI  C 04H     0E 04
       8051 MOV  A,D       7A
LOOP2  8052 CPI  00H       FE 00
          4 JZ   LOOP3     CA 67 80
          7 CALL V ADJ     CD 62 81
          A MOV  A,E       7B
          B SUI  01H       D6 01
          D DAA            27
          E MOV  E,A       5F
          F MOV  A,D       7A
         60 SBI  00H       DB 00
          2 DAA            27
          3 MOV  D,A       57
          4 JMP  LOOP2     C3 52 80
LOOP3  8067 MOV  A,E       7B
          8 CPI  00H       FE 00
          A JZ   NEXT2     CA 78 80
          D CALL V ADJ     CD 62 81
         70 MOV  A,E       7B
          1 SUI  01H       D6 01
          3 DAA            27
          4 MOV  E,A       5F
          5 JMP  LOOP3     C3 67 80
NEXT2  8078 CALL IN33      CD 39 03
          B MOV  H,A       67
          C ANI  F0H       E6 F0
          E CPI  30H       FE 30
         80 JNZ  NEXT3     C2 88 80
          3 MOV  D,E       53
          4 MOV  E,H       5C
          5 JMP  NEXT2     C3 78 80
NEXT3  8088 MOV  A,H       7C
          9 CPI  20H       FE 20
          B JNZ  ERROR     C2 59 81
          E LXI  HDATA3    21 B8 81
         91 CALL PRINT     CD 82 81
MONTH  8094 MOV  A,D       7A
          5 CPI  00H       FE 00
          7 JNZ  NEXT4     C2 A5 80
LOOP4  809A MOV  A,E       7B
          B CPI  00H       FE 00
          D JZ   ERROR     CA 59 81
         A0 ANI  0FH       E6 0F
          2 JMP  NEXT5     C3 B4 80
NEXT4  80A5 CPI  31H       FE 31
```

```
          7 JNZ  LOOP4     C2 9A 80
          A MOV  A,E       7B
          B CPI  33H       FE 33
          D JP   ERROR     F2 59 81
         B0 ANI  0FH       E6 0F
          2 ADI  0AH       C6 0A
NEXT5  80B4 MOV  D,A       57
          5 LXI  HDATA4-1  21 5F 82
       80B8 ADD  L         85
          9 MOV  L,A       6F
          A MOV  L,M       6F
          B CALL PRINT     CD 82 81
          E LXI  HDATA6    21 C0 81
         C1 CALL PRINT     CD 82 81
MADJ   80C4 LXI  HDATA7    21 E3 81
          7 MVI  E01H      1E 01
LOOP5  80C9 MOV  A,E       7B
          A CMP  D         BA
          B JZ   FINAL     CA ED 80
          E CPI  02H       FE 02
         D0 JNZ  NEXT6     C2 DF 80
          3 MOV  A,C       79
          4 CPI  04H       FE 04
          6 JNZ  NEXT6     C2 DF 80
          9 MOV  A,B       78
          A INR  A         3C
          B CALL W ADJ     CP 79 81
          E MOV  B,A       47
NEXT6  80DF MOV  A,B       78
         E0 ADD  M         86
          1 SUI  01H       D6 01
          3 DAA            27
          4 CALL W ADJ     CD 79 81
          7 MOV  B,A       47
          8 INX  H         23
          9 INR  E         1C
          A JMP  LOOP5     C3 C9 80
FINAL  80ED MVI  D00H      16 00
          F MOV  E,M       5E
         F0 CPI  02H       FE 02
          2 JNZ  NEXT7     C2 FD 80
          5 MOV  A,C       79
          6 CPI  04H       FE 04
          8 JMZ  NEXT7     C2 FD 80
          B MVI  E 30H     1E 30
NEXT7  80FD MOV  A,D       7A
          E CMP  B         B8
          F JZ   DATE      CA 11 81
       8102 MVI  H04H      26 04
LOOP6  8104 MVI  A 20H     3E 20
          6 CALL OVT33     CD 00 03
          9 DCR  H         25
          A JNZ  LOOP6     C2 04 81
          D INR  D         14
          E JMP  NEXT7     C3 FD 80
DATE   8111 MVI  D01H      16 01
LOOP7  8113 MVI  A 20H     3E 20
          5 CALL OVT33     CD 00 03
          8 MVI  A 20H     3E 20
          A CALL OVT33     CD 00 03
       811D MOV  A,D       7A
          E ANI  F0H       E6 F0
         20 JNZ  NEXT8     C2 2B 81
          3 MVI  A 20H     3E 20
          5 CALL OVT33     CD 00 03
          8 JMP  NEXT9     C3 34 81
NEXT8  812B RRC            0F
          C RRC            0F
          D RRC            0F
          E RRC            0F
          F ADI  30H       C6 30
         31 CALL OVT33     CD 00 03
NEXT9  8134 MOV  A,D       7A
          5 ANI  0FH       E6 0F
          7 ADI  30H       C6 30
          9 CALL OVT33     CD 00 03
          C MOV  A,B       78
          D INR  A         3C
          E CALL WADJ      CD 79 81
         41 MOV  B,A       47
          2 CPI  00H       FE 00
          4 JNZ  NEXTA     C2 4D 81
          7 LXI  HDATA8    21 F0 81
          A CALL PRINT     CD 82 81
NEXTA  814D MOV  A,D       7A
          E ADI  01H       C6 01
         50 DAA            27
```

```
          1 MOV  D,A       57
          2 CMP  E         BB
          3 JNZ  LOOP7     C2 13 81
          6 JMP  END       C3 70 82
ERROR  8159 LXI  HDATA9    21 F4 81
          C CALL PRINT     CD 82 81
          F JMP  END       C3 70 82
V ADJ  8162 MVI  A04H      3E 04
          4 CMP  C         B9
          5 JNZ  NEXTB     C2 6E 81
          8 MOV  A,B       78
          9 INR  A         3C
          A CALL W ADJ     CD 79 81
          D MOV  B,A       47
NEXTB  816E MOV  A,B       78
          F INR  A         3C
         70 CALL W ADJ     CD 79 81
          3 MOV  B,A       47
          4 DCR  C         0D
          5 RNZ            C0
          6 MVI  C04H      0E 04
          8 RET            C9
W ADJ  8179 CPI  07H       FE 07
          B RM             F8
          C SUI  07H       D6 07
       817E DAA            27
          F JMP  W ADJ     C3 79 81
PRINT  8182 MOV  A,M       7E
          3 CPI  00H       FE 00
          5 RZ             C8
          6 CALL OUT33     CD 00 03
          9 INX  H         23
          A JMP  PRINT     C3 82 81
CHNG   818D CPI  00H       FE 00
          F RZ             C8
         90 PUSH PSW       F5
          1 CALL OVT33     CD 00 03
          4 POP  PSW       F1
          5 ANI  0FH       E6 0F
          7 RET            C9
LSHFT  8198 RLC            07
          9 RLC            07
          A RLC            07
          B RLC            07
          C RET            09
DATA1  81A0   0D 2A 43 41 4C 45 4E 44
              41 52 2A 20 20 20 00 00
DATA2  81B0   20 20 20 41 44 2D 00 00
DATA3  81B8   0D 0A 20 20 20 20 20 00
DATA6  81C0   0D 0A 0A 20 53 55 4E 20
              4D 4F 4E 20 54 55 45 20
              57 45 44 20 54 48 55 20
              46 52 49 20 53 41 54 00
              0A 0A 00
DATA7  81E3   32 29 32 31 32 31 32 32
              31 32 31 32
DATA8  81F0   0D 0A 0A 00
DATA9  81F4   0D 45 52 52 4F 52 20 3F
              0D 0A 00 00
DATA5  8200 1 4A 41 4E 55 41 52 59 00
       8208 2 46 45 42 52 55 41 52 59
              00 00
       8212 3 4D 41 52 43 48 00
       8218 4 41 50 52 49 4C 00
       821E 5 4D 41 59 00
         22 6 4A 55 4E 45 00
         27 7 4A 55 4C 59 00
         2C 8 41 55 47 55 53 54 00
         33 9 53 45 50 54 45 4D 42 45
              52 00
        3D 10 4F 43 54 4F 42 45 52 00
       8245   4E 4F 56 45 4D 42 45 52
              00 00 00
       8250   44 45 43 45 4D 42 45 52
              00 00 00 00 00 00 00 80
DATA4  8260   00 08 17 18 1E 22 27 2C
              33 3D 45 50 00 00 00 00
END    8270   CD 39 03 FE 0D C2 70 82
              C3 00 80 00 00 00 00 00
       8280   45 52 52 4F 52 20 21 0D
              0A 0A 59 4F 55 20 50 55
              53 48 25 44 20 22 30 22
              20 4B 45 59 2E 0D 0A 0A
              49 20 43 41 4E 27 54 20
              52 45 50 4C 59 20 46 4F
              52 20 59 4F 55 2E 0D 0A
              0A 2A 00
```

＊このプログラムはPROM-04サブルーチンパッケージを使用している

フローチャート1
OUT33
ENTRY・DATA ACC……ASCIIコード(bit7＝カナフラグ)
EXIT・DATA なし
83A0 カーソルポインタ(L)
83A1 〃 (H)
OUT33
BC, DE, HL セーブ
キャリッジ・リターンチェック
Acc＝0D
83A0の内容 bit0〜4をリセット
END
LOAD
ラインフィード チェック
Acc＝0A
Acc←20x
PNTSET
カナコードチェック
Acc：負
Acc∧3Fx
Acc∧3 Fx
Acc∨40x
カナデータ・セット
Accの内容をカーソルポインタの示す番地へセットする
Acc←01x
CSMOD
キャリーフラグON
SCROL
END
LOAD
フローチャート2
IN33
ENTRY DATA なし
EXIT DATA Acc…ASCIIコード(bit7＝カナフラグ)
IN10
IN33
BC, DE, HL セーブ
(PNTR)は，カーソルポインタによって示されるカーソルアドレス
PNTSET
Acc←(PNTR)
Accセーブ
(PNTR)←20x
ブランク・コード
B←01x
カーソル用ディレイカウンタ
SCN
KEY ONチェック
Acc＝0
SCN
Acc＝0
Accロード
(PNTR)←Acc
Acc←PORT B
PORTBはKEY・ENCODER・DATA
Acc＝03
03＝CNTRL・C (画面クリア)
CLR33
IN10
英小文字→大文字コード変換
B←Acc
Acc←PORTC
PORTCのbit 1はカナbit
bit 7←bit 1
Acc←AccVB
カナフラグ＋キャラクタデータ
LOAD
フローチャート3
COMOD カーソルメンテナンス
ENTRY DATA Acc…カーソル移動量(2's COMPLEMENT)
EXIT DATA なし
CSMOD
BC, DE, HL セーブ
Acc＝負
カーソルポインタにAccの内容を加え込む
カーソルポインタにAccの内容を加え込む
カーソル・ポインタ (H)＝2？
画面オーバーチェック
カーソル・ポインタを1行分だけ減ずる
LOAD
I/O

図4　アドレス対応表

カーソル・ポインタ

TK-80 RAM内のアドレス83A0と83A1を，カーソルポインタとして使用しています．83A0をLOWER BYTE，83A1をHIGHER BYTEとし，0000～01FFの値となります．

カーソルポインタの値に，TVD-02のアドレス（FE00）を加えたものが，カーソルの実効アドレスとなります．

初期設定

8255のポート設定

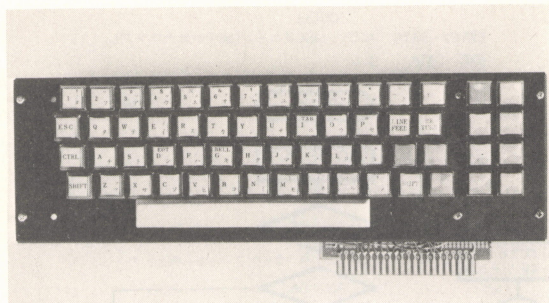
Bポート及びCポート下4bit を入力ポートに設定します．

MVI A, 93H (3E93)
OUT 03H (D303)

ポインタ初期設定

CALL CLR33
(CDB803)

ターミナル・システムを使用する前に，必ず上記の処理を行なってください．

写真2　キーボード

システム・チェック

全インターフェイス完成後，次のプログラムをRAMにWRITEして，ターミナルのチェックを行ないます．

```
8000：MVI    A，93 (3E93)
      OUT    03      (D303)
      CALL   CLR33 (CDB803)
8007：CALL   IN33 (CD3903)
      CALL   OUT33 (CD0003)
      JMP    8007 (C30780)
```

キー入力した文字が直ちにエコーバックして，TV画面に表示されます．画面を消す場合は，

[CTRL] ＋ [C]　を押してください．

◆TVD-02応用例

1．図5のTK-80システムでBASICを！
（写真4）
2．TVD-02とTK-80でCRTデバッガー
（写真5）メモリの内容ダンプ，全レジスタの内容ダンプ，ブレークポインタの設定，ループ回数の監視．
3．各種ゲームとしての応用
一例として万年カレンダーのプログラムを紹介します．これはやはり図5のシステムで，年と月をキ

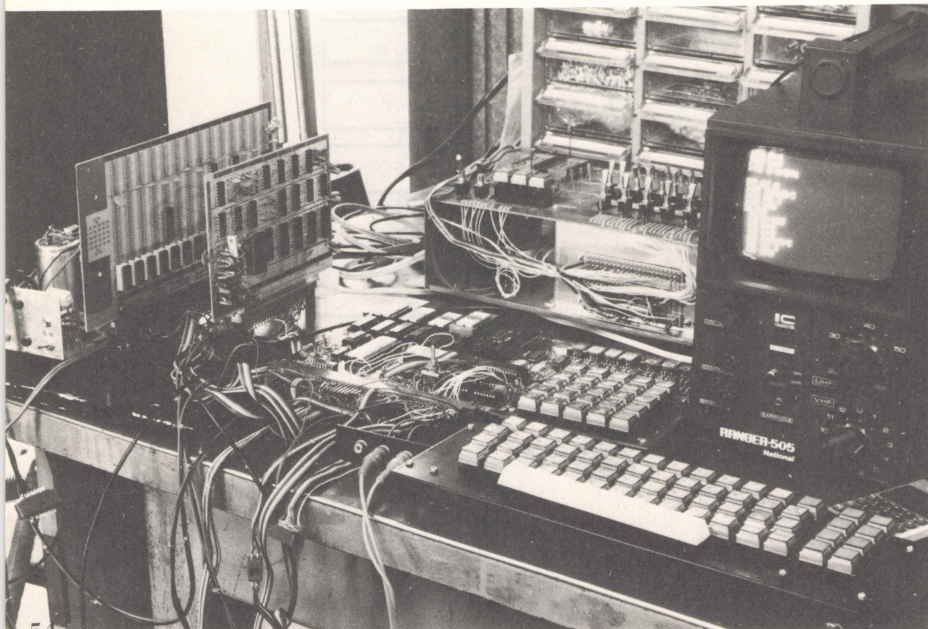
写真3
TK-80　TVターミナルシステム

ーボードより入力すると，その月のカレンダーがＣＲＴ上にディスプレイされるものです．遊び方は8000番地よりＲＵＮをさせ，西暦の年を入力レスペースを入れる，次に月を入れスペースを入れるとその月のカレンダーが表示されます．表示可能年月0000～2099年まで．

図5　TK-80　TVタイプライターシステム

フローチャート4

CLR33

ENTRY DATA なし
EXIT DATA なし

CLR33 → BC, DE, HL セーブ → VRAM全エリア(FE00~FFFF)にブランクコード(20x)をセット → カーソルポインタをゼロクリア → END → LOAD

フローチャート5

SCROL

ENTRY DATA なし
EXIT DATA なし

SCROL → BC, DE, HL セーブ → V-RAM内のエリア(FE20~FFFF)を(FE00~FFDF)に移す → V-RAM内のエリア FFE0~FFFF にブランクコード(20x)をセットする. → BC, DE, HL ロード → RETURN

LOAD → BC, DE, HL ロード → RETURN

フローチャート6

SCAN

ENTRY DATA REG, BC……カーソル・フラッシュ用カウンタ
EXIT DATA ZEROフラグ……1 KEY OFF
……0 KEY ON

SCAN → BC↓ 1 → =0 (N: → Acc←PORT C; Y: ↓) → カーソルポインタで示すメモリを変更する 20→1F 1F→20 (ブランク→アンダーライン アンダーライン→ブランク) → B←20 (ディレイカウンタ初期セット (この定数を変えれば, カーソルフラッシュの周期が変わります)) → Acc←PORT C (PORTC bit0=KEY・STROB) → Acc∧01x → RETURN

フローチャート7

PNTSET

ENTRY DATA なし
EXIT DATA H, L←カーソルポインタの値+FE00

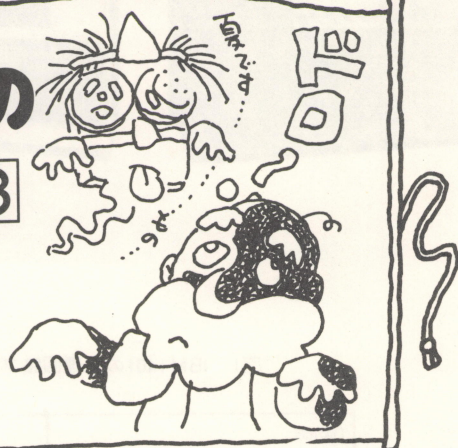
PNTSET → L←(83A0) H←(83A1) → HL←HL+FE00 → RETURN

急げ μCOM 青春はすぐに時間切れ!

I/Oポート

東京(首都圏)地区学生マイコンサークル連絡会(仮称)発足

このたび, 首都圏にある大学生中心のサークルが連絡会をつくりました. 現在の参加サークルは, 電通大MMA, 東大マイコン同好会, 東京理科大無線研, パーソナルコンピューティンググループ, 早稲田大学MISハードウェアグループの5サークルで, ソフト, ハードにわたる情報交換, ミーティングを行ない親睦をはかっていきます. また, 8月中にバスの共通規格を検討するとともに, ソフト(モニター, ローダ, データ・フォーマットetc)についても規格を検討する予定です. 各大学の文化祭終了後に本格的な運営にはいりますが, 他に興味をお持ちの方は是非御連絡下さい.

《連絡先》
〒230 横浜市鶴見区矢向4-31-18 土田米一
☎(045) 572-6977

この昔話の第一回目（5月号）に，ＩＢＭ650の物語りをとり上げたところ，この記事を見て下さった多くの方々から，是非連載にしろという，はげましと共に『次は当然1401でしょうな.』という予想というか要望（何しろ書くのは当方なので）を多くいただきました.

そうは問屋がおろさぬ．というわけで二回目はソフトの話にしたが，実用コンピュータの歴史（筆者の書いているのは，技術開発史ではありません．これはあちこちで開発年表などの形でとり上げられていますし，このような事を調査する専門家もあるようですからそちらにゆずります.）を連載形式でとり上げていくとなれば，いかにヘソ曲りを自認する筆者としても，このＩＢＭ1401ばかりは，絶体にバイパスできないコンピュータです.

それほどに，このマシーンの存在は大きなものがありました．そこで今回は，大方の予想通り（クヤシイけれど）この当然に書かれるべきマシンの話といたします.

◘世界を征服した1401

第1回目で話したように，ＩＢＭ社は650型の成功によって，当時数十社もあった競走会社を一歩抜き，一応トップには立ちましたが，まだまだその地位は不安定なものと見られていました.

事実，コンピュータ業界の草分けであり，かつて首位でもあったユニバック社は，650対抗機種として，いちはやくUSSC（ユニバック・ソリッド・ステート・コンピュータ）を市場に送り，市場奪回を目指して猛攻撃を加えていましたし，日本のコンピュータ・メーカーも，完全トランジスタ化（なんともなつかしい言葉です）のマシンによって対抗しようと，新鋭機開発に力をかたむけていました.（650のトランジスタ化です.）

こんな状況の所へ，突序として現われたのが本機です.『それは彗星の如くに出現し，またたくまに世界を席巻した.』（名調子だね，エヘン）といっても過言ではありません.

この機械の発表は1959年でしたが，本格的な出荷が始まったのは60年になってから，従って60年代前半を代表するコンピュータと考えてよいでしょう.

1964年頃，全米のコンピュータは約18,000台，このうちで１万台以上が本機であったというのですから，ものすごいとしか，いいようがありません.

参考迄に書き加えると，64年に日本におけるコンピュータ設置台数は，1,000台の大台を越しました．また米誌フォーチュンの発表する，有名な“全米企業ランキング”というのがありますが，これによれば，64年度にＩＢＭ社は第９位（前年は18位）と始めてシングル入りを果しています．これもこの1401の大成功の結果である事はいうまでもありません.

むろんＩＢＭ社のマシンは1401型だけではなく，この上に7070とか7090といった，いわゆる7000シリーズと称する中大型機も作っていましたが，この中では一番多く売れたという7070でも600台強で，他社に比べれば多い方（日本のメーカーが聞いたら失神するわナ!!）ですが，本機に比べれば文字通り桁違いというやつで，問題にはなりません.

そんなこんなで，本機に関する記録やエピソードなどは書くとキリがありません（興味のある方は60年代前半のアメリカの雑誌でも調べて下さい.）ので，こんな事にしますが，ただこの1401機こそは，『史上始めて1万台を突破したコンピュータ.』

『コンピュータにおけるＩＢＭの王座を確立した.』

という二つの点を記憶にとどめておいて下さい.

◘キャラクタ・マシンとは

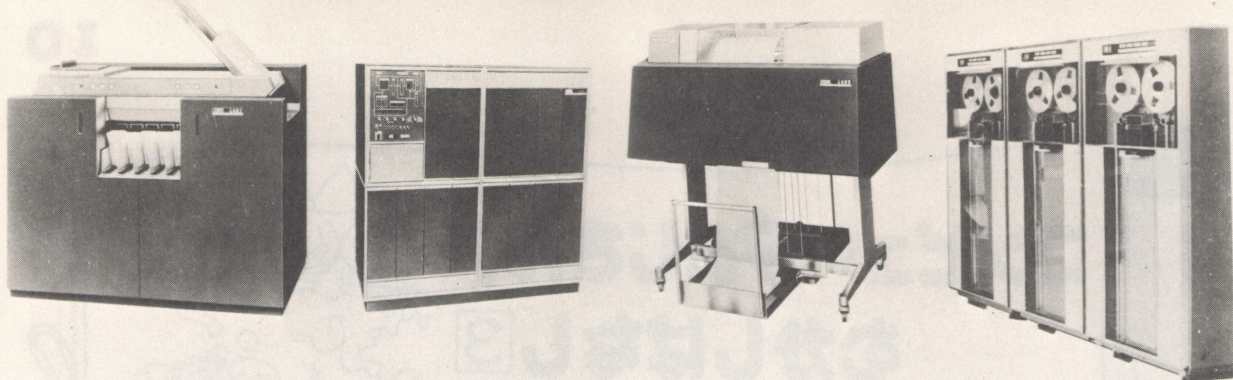
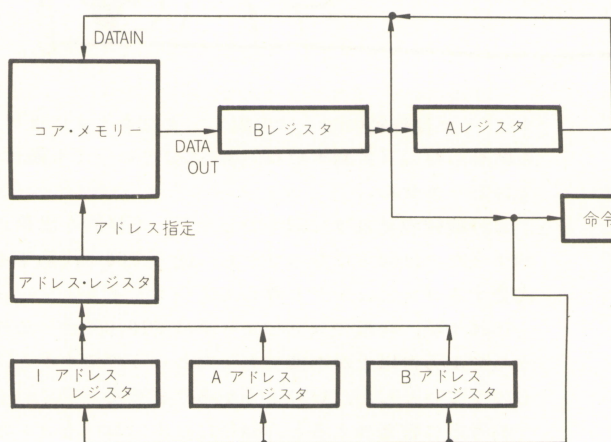
それでは，このものすごいマシンは，どのような考

▲ＩＢＭ1401システム

図1　IBM1401の論理回路ブロック図

え方でどんな規模のものだったのでしょうか．

まず技術思想的に最大の特徴は“キャラクタ・マシン”という点です．

従来のコンピュータが，データー（命令も）を操作したり，メモリーに記憶させたりする単位として，ワード（長さはいろいろ）の考えを持っていたのに対して，1401はキャラクタつまり文字を単位としました．これ以後，従来の考え方を“ワードマシン”と呼んで区別するようになりました．

この文字は8bitで表わすように構成されていました．「なんだそれなら，現在の8bitマイコン等でおなじみのものだ,“バイトマシン”といっても同じ事ではないか？」と思う人もあるでしょう．

しかし，これはチョット違います．実をいえば下のように，割当てられているのは8bitですが，この内で1bitはパリティチェック用，さらに1bitはワードマークと呼んで，この各文字で構成する個々の単語の切れ目を示すマークです．

パリティビット↓　↓マークビット

P	M	d_5 d_4 d_3 d_2 d_1 d_0

したがって，厳密な意味で内容を示すのに使われるのは6bit分で，これでＡ～Ｚの文字と０～９の数字や各種の記号を表現します．

コード系は省きますが，表現しうる文字はＩＢＭコードに表現しうるもの（いわゆるホレリス文字）の全てであって，今日のASCII体系とは異ります．

メモリー内では，この文字単位に番地がつけられていて，操作もまた文字を単位として行われます．そこで“キャラクタ・マシン”と呼ばれるわけですが，文字の上位概念には単語があるわけで，本機の場合この単語の長さは，ワードマークのつけ方によって任意に設定できる事から“可変語長マシーン”とも呼ばれます．（対応語は“固定語長マシーン”です）

◇ハードウェアの構成

メモリーはコアを使用し，その容量は最低で1,400字（これが機械の名前になったのです．）から，2,000　4,000，8,000，12,000，16,000字と増設可能になっていました．考え方は10進法なので，今日のマシンのように4Ｋ＝4,096などという，非人間的なものではなく4,000はジャスト4,000字です．

回路は全てトランジスタで構成され，概略を示す

と図１のようになっていました．

この中のＡＢ両レジスタと命令レジスタは文字の入れもので（1文字だけ入る），アドレスレジスタとしてある４本のレジスタは，それぞれ３文字のアドレスレが入ります．

図では演算論理回路（ＡＢレジスタの出力が，これを通って，メモリーへと入る）が省略してありますがそれにしても簡単なもので，どこか今日のマイコンの内部構造に似ています．（ただし，計算も十進法です．）

ＩＢＭは文字単位のプロセッシングを考え出す事によって，大量の事務処理に使われ始めたコンピュータを，極めて使いやすいものにする（事務用にはワードマシンは使いにくい）と同時に，ハードの簡略化をも達成するという，一挙両得の策を取ったのです．

“THINKING” をくり返すワトソン会長の精神が見事に生かされた好例といえるでしょう．

もちろん欠点もあります．文字単位などという，細かな操作単位で，何かの処理をしようとすれば，何回も何回も同じような事をくり返さなければならず，このために時間がかかる事は否めません．

図１を見て考えて下さい．まず命令のフェッチにしてからが，Ｉアドレスレジスタ(カウンターを兼ねる)を次々とインクレメントしながら，メモリーの番地指定をし，この指示によって命令語（これも可変長）の第１文字（ＯＰ）を命令レジスターに送り，第２文字以下を，それぞれ３文字づつ，ＡとＢのアドレスレジスタに送るという動作をせねばなりません．つまりフェッチだけで，このグルグル廻りを７回くり返すわけで，この命令の実行はまたグルグル（ヤレヤレ書くのもシンドイな!!）．

そんな次第で，メモリーアクセスタイムこそμｓオーダになった（ドラムに比べれば３桁以上速い）ものの，一つの命令を実行するためには，短くても100μsオーダーはかかった.(何しろ素子はトランジスタとはいってもゲルマちゃんですからね.)

これではプログラムの方で，またまたグルグルの多い科学技術計算などには，とても使えない事になる．

しかし事務計算なら一般的に内部演算は簡単で，主として問題になるのはＩ/Ｏ（アイ・オーですぞ!!）です．

カードリーダにせよ，ラインプリンターにせよ最高速なものが，ｍｓのオーダーだから，ちょうどバランスがよいわけです．そしてコストは下げられます．

こんな事から，作表を中心とするバッチ処理主体の当時の事務用コンピュータとして大量に売れたのです．

◘価額はどれくらいか

それでは次に，ハードの簡略化によって，当然安くなった価額がいくらか，という問題ですが，正直にいってこれが大変難かしいのです．

なぜかというと，この1401型は（売れたからもあったのだろうが）1400シリーズと称し，極めて多種の付属装置を保有し，どれが標準かは容易に規定しにくい．ではＣＰＵだけならと問われても，これにも実は前述のように，各種のメモリー容量のものがあり，なおその上に多くのオプション回路（インターフェイスではなく，ＣＰＵとしての回路．例えば乗算回路などはオプション）が用意されていて，これまたどの辺が標準か判らないのです．

しかし，何も指標がなければ，今日のものとの比較やマイコンと比べて（そして読者にボクチャン幸せと思って）頂く手がかりがない．そこであえてつけると，ＣＰＵの 4,000字のものでベーシックなものなら,100万円余りでした．（もちろん，月当りのレンタル料です）

これにカード・リード・パンチャを１台，これも多種類ありますが，1402型を選ぼう．そしてプリンターには600字/分の1403型を採用する事にする，これで一応カード中心システムが組めますが，この程度のものなら170万円/月ぐらいに収まります．

今から振り返ってみれば，この性能のものでこの価額は決して安いとはいえないでしょう．（特にＣＰＵは).しかし当時としては大変に格安でお得用な機械だったのです．

もっとも,本機を導入した日本のユーザーの多くは，以上のような基本構成に近いものは少なかったようで，たいてい４台以上のＭＴ装置か，1405型磁気ディスク装置（25枚，50面の大型ディスク，その代り遅かった）などを連結したものが多く，レンタル料も 400万程度のものが多かったのですが，それでも他社の同級機よりは安かったものです．

◘1401の命令語

ある特定のコンピュータが，どのような性格を持つものであるかは，アーキテクチャとともに機械語の命令をよく見れば，大略のところは判るものです．

そこで次に本機の命令語（インストラクション・セットと呼ばれる）を一覧表にしてお目にかけましょう．（表１）

読者の中には『おやおや，これはアセンブラの簡単なものではないのか？こんな機械語なんてあるのか.』と思われる方もあるかも知れません．

しかし，本機は文字マシンである事を想い起して下さい．メモリーの各番地に入っているものは，それが命令語であろうが，データであろうが，全て文字なのです．（それにしても，うまく割当てたものです.）

これがよく頭に入ったら，キャラクタ・マシンに関する限り，ビットイメージとか，ヘキサデシマルなどという奇怪千万な概念は全く必要ないのです．

表1　命令表

IBM1401命令コード表

区分	コード	命令		区分	コード	命令	
データ制御命令	A	加算		論理制御命令	B	分岐	
	S	減算			V	条件付分岐	
	@	乗算	※		C	比較	
	%	除算	※		N	何もしない	
	⇧	クリヤーして加算			・	停止	
	○̄	クリヤーして減算			W	ビット比較分岐	※
	M	AをBに移す		入出力制御命令	1	カードを1枚読む	
	P	レコードマークまで移す			2	1行プリント	
	Z	ゼロサプレスして AをBに移す	※		3	プリント&リード	
	D	Aの数字部のみBに移す			4	1行パンチ	
	Y	Aのゾーン部のみBに移			5	リード&パンチ	
	E	移して編集			6	プリント&パンチ	
	L	ワードマーク付で移す			7	プリント・リード&パンチ	
	,	ワードマークを付ける			8	パンチフィード	※
	□	ワードマークを取る			9	リードフィード	※
	／	メモリークリヤー			K	カードスタッカ選択	
	X	頭のスペースをゼロにして移動	※		F	改行	
	Q	(Aレジスタ)→A	※				
	H	(Bレジスタ)→B	※				

以上全36種
ただし※印の命令は該当するハードウェアオプション回路が必要である.

機械語そのものに関する限り，今日迄に出現したあらゆる方式のコンピュータの中で，最も判り易い方式であるといえるでしょう.（この点については筆者は未だにノスタルジアを感じています.）

表1は命令語の中の命令コード（OP部ともいう）を表す文字であり，個々の命令は（中にはOP部のみという命令もありますが）一般に，このあとに3文字の番地部が一個か二個つき，さらに場合によってその後ろに1文字の修飾子（d部ともいう）がつきます．つまり命令語も可変長で，最短1文字から最長7文字の各種の長さのものがあるわけです．

例を示しましょう．

1) | A | 5 | 2 | 0 | 5 | 6 | 0 |

OP　Aアドレス　Bアドレス

加算命令である．
（Aの下にバーがあるのはワードマークを示す）

520番地の内容を560番地に加え次に519番地の内容を559番に加え以下同様にBアドレスにワードマークが現われるまでつづけよ．
Aアドレス側に先にワードマークがあった時はゼロを加えつづけよ．

2) | L | 7 | 8 | 0 | 4 | 5 | 0 |

ワードマーク付転送命令
（転送の長さはAアドレス側で決定される）

780番地の内容を450番地に移し次に779番地の内容を449番地に，以下同様にAアドレス側にワードマークが現われるまでつづけよ，ワードマークがあったら，マーク付で移して終れ．
Bアドレスの途中にあるワードマークは消せ．

3) | 1 | リード命令

カードを1枚読み，001番地から080番地に格納せよ．
この時にもとからあるワードマークは変化しない．

4) | 1 | 3 | 7 | 0 |

カードリード&ブランチ

カードを1枚読み，次は370番地から実行せよ．

ここでは判り易く，番地部にはすべて数字を入れてありますが，実は3文字の数字で表現しうる番地は考えるまでもなく，000～999の1000番地分です．これ以上はどうして指定するのか？

ここにも文字が登場します．全てを示す事は（たいして意味がないので）省きますが，これも例を示すと

T00は1300番地
U00は1400番地
M23は2423番地
H45は3845番地

といった具合です．このように最上位に各種の文字を使う事によって，3文字で3999番地迄の直接指定をします．これ以上はINDEXレジスタを使うのですが，これも省略します．

◘入出力エリヤは

さて上記の**例3**では，ただカードを読めとのみで，メモリーのどこに読みこむのか？という規定はしていません．

これはどういう事かといえば，基本装置の入出力エリアは，始めから決めてあったのです．つまり

カードインプットエリヤ	;001～080番地
カードアウトプットエリヤ	;101～180番地
プリントアウトエリヤ	;201～300番地
(132字/行プリンターの時は)	;201～332番地

という具合です．このためにこのエリヤは自由に使う事ができない反面，プログラムは大変簡単になります．（何の事はない，今日マイコンマニヤがさも大発明のように，“VIDEO-RAM” などといっている考え方は，

20年近く前からあったのだ，残念デシタ!!）

もっとも，こんなにカンタンに行くのは基本的カードシステムの場合に限られ，MT装置など使おうとすると，とたんにヤヤコシクなります．

OP	A部	B部	d部
M	% U ×	× × ×	R

MTリードライト命令
（d部Rがリード，Wはライト）

機械語では移動命令で示される．A部の下位桁で号機を示し，B部でメモリーリードエリヤのMSDを示す．

命令そのものは上のように，一語で片づきますが，レコードの長さが，どうのこうのというような事や，バッファーがいるなどから，どうしても，かなり大きなエリヤを必要とする事になり，最低 4,000字，欲をいえば更にエリヤが欲しくなり，この制御が面倒だから既製のコントロールルーチン（IOCSですな）を使う事にすると，更にこのルーチンのエリヤも必要となり，磁気ディスクも使いたいとなれば，更に更に高度なルーチンが入用………これではたちまちメモリーがパンク（例え最大の16,000字ものを使っても）する事になり，それではこういうコントロールプログラムをディスクに退避させておいて必要に応じてメモリーに展開して使えばよい．という事になりますが，このためにも，そういう各種のルーチンを制御する，上位の制御プログラム（エギゼクティブルーチン）がいるというわけで，かくして○○システムなどという，プログラムのお化けが出現する（OSの始まりです.）のです．

◇1401の残した遺産

さてこのように，世界を席捲した1401にも，上述のように，I/Oの多様化とプログラム技法の高度化とから，退場の時がやって参りました．

この幕を降ろしたのも，生みの親であるIBM社自身であったのは，御立派という以外に表現しようがありませんが，この事については，また何回か先の続編でふれる事にします．

ただここでは，その後の1401，つまりこの系統の文字マシンはどうなったかを述べておきましょう．

IBM社は，この1401用に言語システムとして，最低構成から使えるSPS－1（シンボリック・プログラミング・システム，2パスのアセンブラです）を始め，SPS－2，AUTO－CODER（MTを使うマクロアセンブラ）を先ず用意し，つづいてお家芸のFORTRAN，COBOLを始めとする各種コンパイラを市場に送りましたが，やはり簡単な構成を主眼に設計された本機に対して，あまり高度なソフトウェア・システムを乗せるのは無理があったようです．特に我が国では随分こった使い方を考え出す名人があちこちにいて，かなり厄介な処理をこなしていた所もありますが，見ているとMTばかりが，クルクルと忙がしく動くばかりで，結果が出るまで随分時間がかかりました．

IBM1440電子計算組織

これを補強する意味からでしょう．IBM社では翌60年，早くもこのシリーズの上位機種として1410型を発表（メモリー容量を増すと共に高速化したもの）しましたが，意外に不振でした（高かったのです）．

つづいて62年には，1401より安く，1311型磁気ディスクパック（これは今日まで，ディスクパックの主流として続いている，有名な装置です.）を強力な補助記憶装置として持つ1440型を発表します．

この機械は我が国で作られた（日本アイビーエム）事もあって，日本のユーザーにも多数設置されました．さらにこの若干あとに発表された1460型をもって，この1400シリーズは終るのですが，この文字マシーンは（簡易型としては）あまりにもよく出来ていたためでしょう，その後にハネウェル 200シリーズに受けつがれ（今日でいうセカンドソースですな.）て結構ユーザーを確得しました．

また我が国ではハネウェルと提携した日本電気の手によって国産化され，NEC2200シリーズとして発売され，これまたヒット（発表は64年）．

これによって日本電気は，しばらく国産6社の主位に立ちました．（1966，67，68年頃の事です）．

ただ今日からふり返ってみれば，この文字マシーンの好調が，かえってその後のバイトマシーンへの移行を遅らせ，早期に切替えをした富士通，日立の両社に抜かれる事になったのは，歴史の皮肉といえるでしょう．（最近になって日電のまき返しが始まってます）

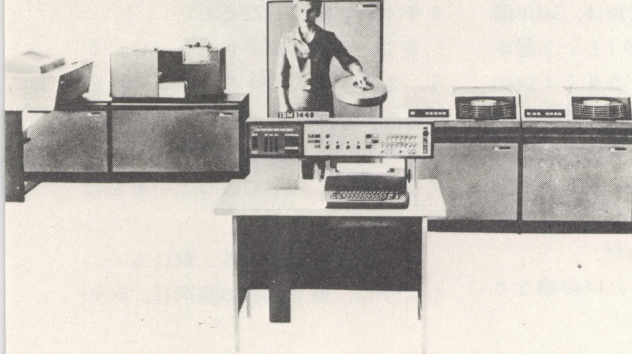
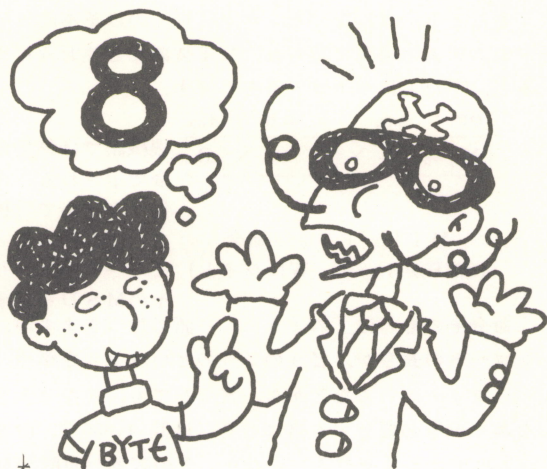
このように1401の残した足跡は功罪共に大きなものがあります．

◇参考文献

F.P.FISHER & G.F.SWINDLE: *Computer Programing Systems*

この本はIBM 1400シリーズのプログラム・システムの開発者によって書かれたもので，特に1401とそのプログラムが中心になっている．

キャラクターマシンの解説書として定評があり，下記の翻訳書も出ている．有名な本だから学校図書館などにあるだろう．

浦昭二訳：“電子計算機プログラミングシステム”培風館 1967年6月初版

ミスターXの プログラム何でも相談室 5

《今月の質問》数当てゲームのプログラム

今月の相談はどれにしようかな．たまには女の子からも質問がこないかなー．よし，今月はこれにしよう．

> **Q** マイクロコンピュータをやり始めたばかりの者ですが，プログラムを教えていただきたいのですがそのプログラムは，次のような数当てゲームのものです*……（中略）このゲームのBを，マイコンにやらせたいのです．つまり，マイコンに数を当てさせるのです．（後略）
> (大阪　T.K.生，ＴＫ－80使用)

T.K.君．きみはもう少し文章の練習をしたまえ．ミスターXがこのゲームを知っていたからいいが，きみの説明では何が何だか，さっぱりわからないよ．

それにしても変ってるね，きみは．普通はこういうゲームはね，Aの方をマイコンにやらせるもんだよ．つまりマイコンに相手をさせて，自分が楽しむのだ．それを楽しみの多い方をマイコンにやらせて，つまらない方を自分がやろうというんだからね．でも，ほかの人にも，プログラムは面白いだろうから，今月はキミの質問でいこう．

こういうものはまずゲームを研究することから，始めるんだ．小中学生の諸君には，意味のわからない言葉がでてきたら，お兄さんか，お姉さんに聞いてくれたまえ．

まず，10ケの数字から，４ケを選んで並べる並べ方，これを，これからただ『並べ』ということにしておこう．この並べは，全部で，

$10\times 9\times 8\times 7=5040_{10}$
$=13B0_{16}$

だけある．エッ，何でそんなところに16進数がでてくるって？マイコンしようなんていう人はね，何でも数字がでてきたら，16進数にしてみるものなんだ．

さてゲームにもどろう．始めに，Aは，５０４０個の並べの中から，１ケ選んで問題とするのだ．

Bは，やはり5040個の並べの中から１ケ選び質問する．これに対して，(４，４)から(０，０)までの14ケの返事のうち１ケが返ってくる．エッ15ケだろうって？よく考えてみたまえ，(４，３)という返事はないんだよ．運がよければ１回で(４，４)という返事が返ってくる可能性もあるが，この確率は1/5040だ．大体は，(１，０)とか(２，０)とかいう返事が返ってくる．その割合は，5040個の並べのうち，(１，０)という返事が返ってくるのが，一番多くて1440個，(２，０)という返事が返ってくるのが1260個ある．つまり，１回目の質問で，(１，０)という返事が返ってきたら，答は，この1440個のうちの一つだとわかるのだ．

２回目の質問で，この1440個なり1260個なりが，もっと減る．これを繰返していって，最後に１ケだとわかったときには，その並べをいえば，確実に（４，４）という返事が返ってくる．それまでの質問の数が勝負なのだ．

このプログラムでの一番の問題点は，『なるべく早くあてるためには，どういう質問をしたらいいのか』ということだ．だが，この質問の仕方を教えてくれ，というのは麻雀の必勝法を教えてくれ，というのと同じに，無理な相談なのだ．ここは，きみの経験で，なるべくよさそうな方法を考えたまえ．

ただちょっと学のあるところを見せておくと，可能性の残っている並べのうち，14ケの返事のうち，i番目の返事の返ってから並べの数をa_iとすると，

$$\sum_{i=1}^{14}(a_i\times\log a_i)$$

をなるべく小さくするような質問が，有利なことが多い．その理由を知りたい人は，大学の数学科へ行って，４年ぐらい勉強したまえ．

さて，それではプログラムにいこう．さて，ＴＫ-80 のマニュアルを出してくれたまえ．ＴＫ-80のことがわからないとこまるからね．マニュアルはキットの中に入っているものだ．

それでは本題に入る．前にもいったように，第１回目の質問は，5040

＊数あてゲームの説明はP.65にあります．

個の並べのうち，どれを使っても，（４，４）という返事の返ってくる確率は，1/5040だし，（１，０）という返事の返ってくる確率は，1440/5040だ．つまり，最初の１回は，どの並べを使っても，確率的には，同じことなのだ．２回目の質問でも，１回目に使わなかった数字は，どれを使っても結局同じになる．つまり，始めて質問に使う数字は，どれから使っても確率は同じなのだ．では，０，１，…，９の順に使ってもいいか，となるとそうはいかない．そんなことをすれば，出題する方が，かぎつけて大きい数字ばかり使うようになる．そこで，ＲＡＭに，

$$X_0, X_1, \cdots\cdots, X_9$$

という10バイトのエリアを用意し，質問を始める前に，０から９までの数字を，このエリアに，ランダムに入れておく．そして使うときには，X_0，X_1，…の順に使うのだ．そうすれば，第１回目の質問は，必ず，

$$X_0\,X_1\,X_2\,X_3$$

になるし，２回目以後は，例えば，

$$X_0\,X_4\,X_5\,X_6$$

のように，X_4以降を，やはり，サフィックスの小さい順に使っていくことになる．この方法のよい点は，プログラムの中で，並べを考えるときには，

$$X_i\,X_j\,X_k\,X_l$$

を考える代りに，

$$i\ j\ k\ l$$

を考えておいて，質問するときだけX_iなどに直せばよいことだ．

それでは，**フロー１**を見てくれたまえ．これが全体フローだ．このフローには山が三つある．一番の山は質問の内容をきめるところ．さっきもいったように，これは，自分で考えるんだよ．あとの二つは，①数字をランダムに並べる方法と，②並べをリストアップするのに，どうやって，記憶容量を小さくするかの二つだ．きみが大ブルで，ＲＡＭの10Ｋもあるマイコンを使っているなら②の苦労はないんだよ．

フロー１　全体フロー

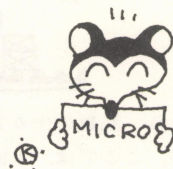
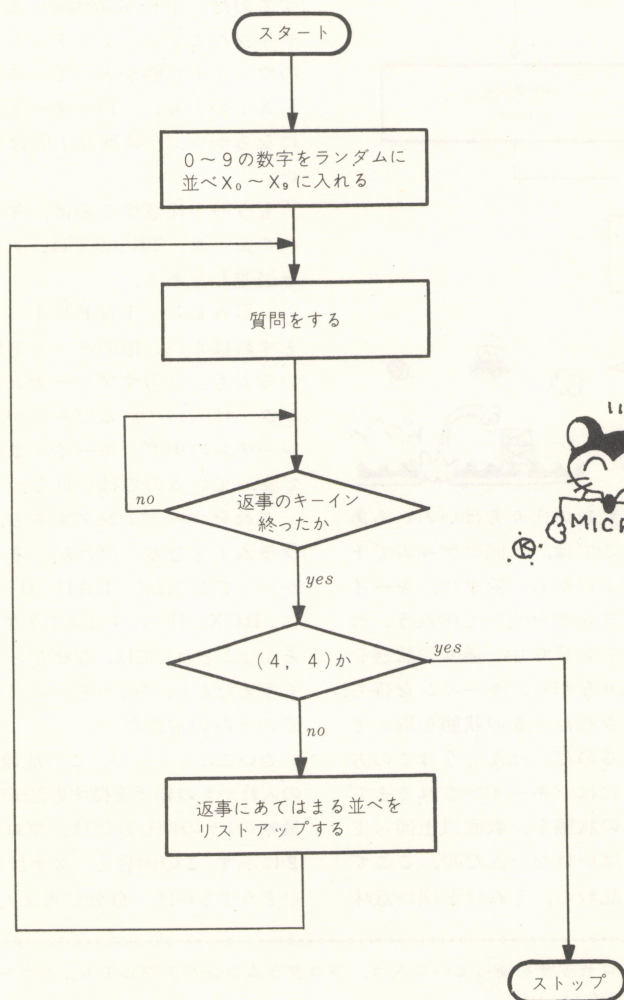



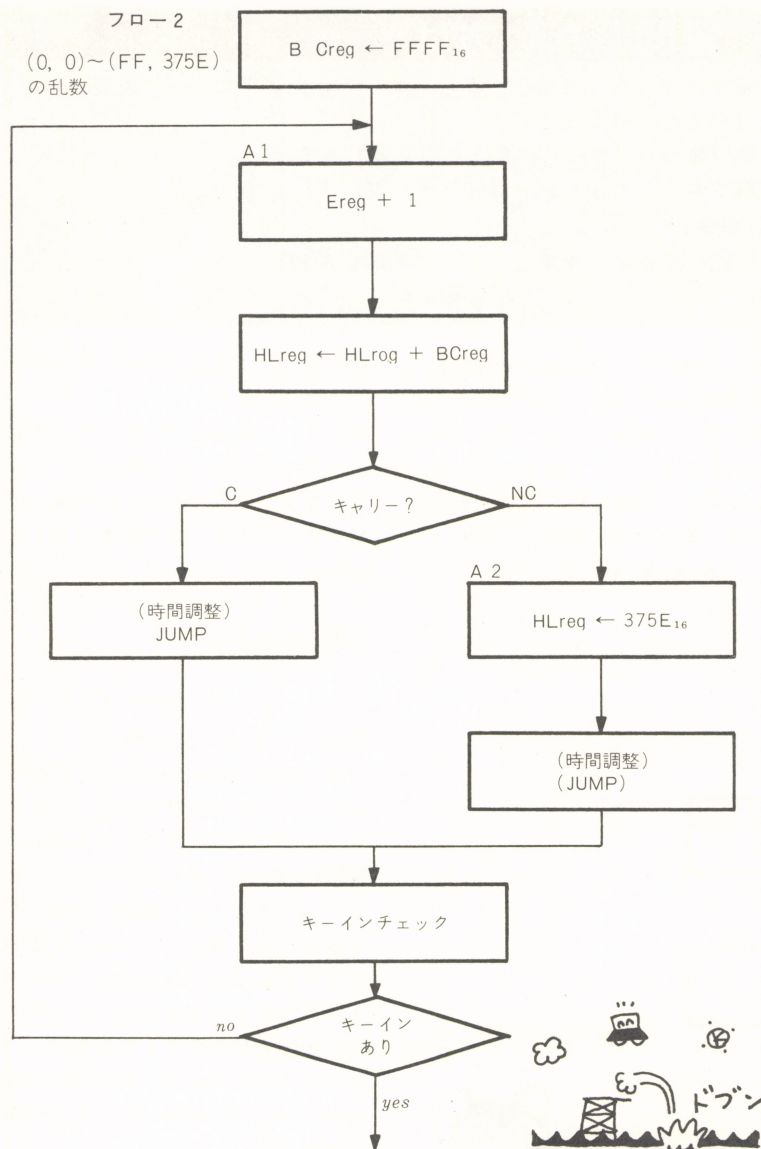
図１　１０！の計算

$$
\begin{aligned}
&10\cdot9\cdot8\cdot7\cdot6\cdot5\cdot4\cdot3\cdot2\cdot1\\
&=5\cdot9\cdot1\cdot7\cdot3\cdot5\cdot1\cdot3\cdot1\cdot1\times2^8\\
&=(9\cdot7)\cdot(5\cdot3)^2\cdot16^2\\
&=(4\cdot16-1)\cdot(16-1)^2\cdot16^2\\
&=(4\cdot16^3-9\cdot16^2+6\cdot16-1)\cdot16^2\\
&=3\cdot16^5+7\cdot16^4+5\cdot16^3+15\cdot16^2\\
&=375\mathrm{F00}_{16}
\end{aligned}
$$

質問したい方は……

- プログラムで解らないこと，コーディング・エラーの修正etc. 何でもけっこうです．
- プロセッサは一応8080Aを中心とします．
- プログラム・リストは50ステップ以下にしてください．

《宛先》

〒151 渋谷区代々木２－５－１羽田ビル403
I/O 編集部「ミスターX」係

①の方からいこうね．まず０〜９の数字をどういう順でもいいから，X_0〜X_9にいれておく．それから適当に入れかえようというんだ．この入れかえ方は

１０！＝３７５Ｆ００$_{16}$

通りある．おいおい！きみ！そこで何をしてる！なに １０！を計算してる．もっと要領よくやれよ！**図１**をみたまえ．

ところで，この入れかえをするには，３７５Ｆ００$_{16}$通りに変る乱数がいる．乱数の作り方はいろいろあるが，ここでは，１回のゲームで１ケあればよいから，簡単に，キーインのタイミングを使って作ろう．つまり，３７５Ｆ００$_{16}$通りの状態を次々に取りながら，キーインを待ち，キーインされたときの状態を取って乱数とするのだ．ほんとうはこの方法を取るには，キーインされるまでにすべての状態を，数回以上回っていなければいけないんだが，ここでは，この乱数に，それほど深い意味はないから，いいことにしておこう．

ここで，０〜３７５ＥＦＦ$_{16}$までの乱数と書かずに，３７５Ｆ００$_{16}$通りの乱数と書いた．それは，

（０，０）〜（ＦＦ，３７５Ｅ）

の乱数を作るつもりなのだ．**フロー２**をみたまえ，始めに（０，０）をセットして，このフローの部分へ入ると，（１，３７５Ｅ），（２，３７５Ｄ）……と進み，（ＦＦ，３６６０）の次が（０，３６５Ｆ）となる．１００$_{16}$と３７５Ｆ$_{16}$とが互いに素だから，これで，

１００$_{16}$×３７５Ｆ$_{16}$

通りのすべての組み合せを，順番に巡る．フローの中に，時間調整と書いたところがあるだろう．これはね，３７５Ｅ$_{16}$の出る確率を他と同じにするために，右と左の時間を同じにするのだ．ＴＫ-80の場合，各マシンサイクルごとに，１ステートメントのウェイト状態が入って，ＪＭＰとＬＸ１がともに，13ステートメントになるから，(注) 特別な工夫はいらない．

もう一つ注意するのは，キーインのチェック，TK-80では，レジスタを退避したあと，

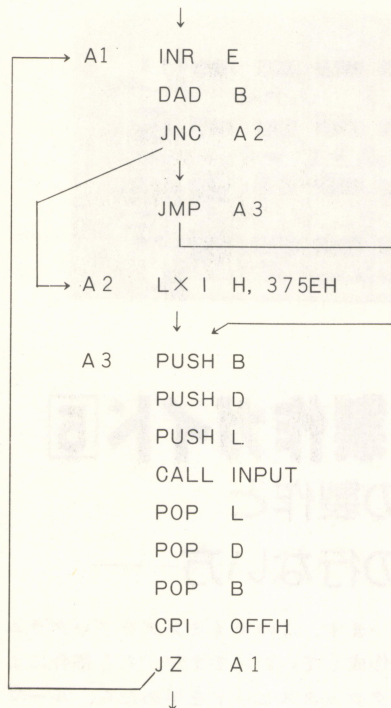
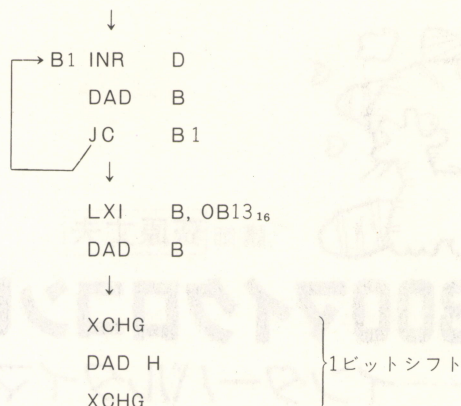
ＣＡＬＬ　ＩＮＰＵＴ

とすればよい．他のキットを使っている人も，このサブルーチンは，モニターに入っているだろうが，サブルーチンの中で，キーインされるまでまっているのではいけないよ．

これをコーディングすると，**プログラム１**となる．だれだ，そこで何かいってるのは！ DAD B の代りに DCX H にするって？ダメだよ，そんなことをしては，なぜだって？よく考えたまえ．ちょうどいい．来月までのきみの宿題だ．

さいごにもう一つ，この乱数で，X_iの入れかえの相手を探す方法がいるね．最初の1ケの探し方だけ，**プログラム２**に示す．この内容と，２ケ目から先をどうするかは，自分で考えたまえ．

(注) TK-80に使っているμPD8080A での値，他のプロセッサを使っている人は，プログラムコンパチブルでも，ステートメント数まで同じとは限らないから，各自調べられたい．

プログラム1

プログラム2

数あてゲーム説明

このゲームは，出題者(A)に対し，解答者(B)が，質問を繰返し，できるだけ少ない質問の回数で，Aの出題した数をあてるものである．

㋑ Aは出題する数を，つぎのようにきめる．すなわち，0~9の10ヶの数のうち，4ヶを選び，この4ヶの数をすきな順に並べる．例えば，5784となる．この数をBがあてる．

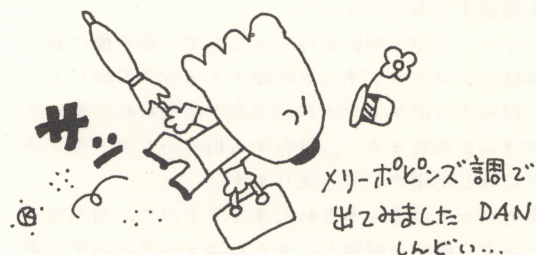
㋺ Bは，例えば4759というように，やはり，4ヶの数を並べて，これを質問としてAに示す．

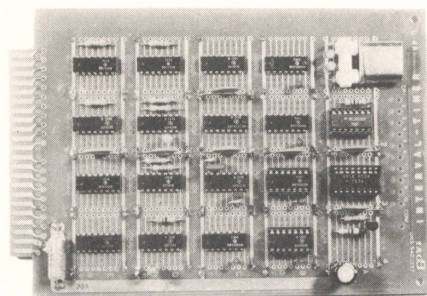
㋩ Aは，出題した数と，質問された数とを比べ，それぞれに，共通する数がiヶ，そのうち位置まで同じものがj個あったら，(i，j)と返事をする．

例えば，出題が5784，質問が4759ならば，共通する数が574の3ヶ，そのうち7だけが，どちらを見ても2番目で，位置まで同じだから，(3，1)というように返事する．㊟

㊁ Bは，質問を繰返し，出題と全く同じ数を質問して，(4，4)という返事が返ってきたら，1ゲーム終り，それまで質問の回数が少ないほど良い．

注：返事の仕方は，位置と値の当っている数の数jと，値だけ当って位置の異なる数k($=i-j$)を用いて，例えば(3，1)の代りに1H2B(Hがj，Bがkとか，ブルカウカウ(ブル＝BULL，j個並べる，カウ＝COW，k個並べる)とか，答える方が一般的だが，ここはT．K君の答え方を用いた．

誌上マイコン学習塾

テスターだけで作る

講師 荻原丈夫

M6800マイクロコンピュータ製作ガイド⑤

——インターバルタイマーカードの製作と総合テストの行ない方——

◇インターバルタイマーについて

マイコンで制御装置を作るとわかるのですが，さまざまなタイミングを作り出したい事が多いものです．プログラムのループで作る事も可能ですが，不便な事も多く，専用のタイマーを使った方が確実で有利です．

筆者がこの必要を感じたのは，'76年の４月頃の事です．秋葉原のジャンク屋で手のひらに乗るくらいの小形プリンタが売られてました．このかわいらしいメカニックに興味をひかれて買ったのですが，この種のプリンタは動かすのが難かしいのです．

ハンマーを駆動する小さな電磁石が印字桁数分15個とモータそれに活字ドラム位置検出用電磁気ヘッドが2個という単純な構成なのですから，買ってもまともに動かせるあてはありません．ＴＴＬで制御回路を組むなら大規模なものとなり，プリンタ制御にしか使えない回路に大金を使うのもいやでした．そんな時に運命のいたずらかトランジスタ技術別冊『つくるコンピュータ』を本屋で見つけ，マイコンでプリンタ制御するのもおもしろそうだと思い，さっそくMYCOM-4の製作を開始する事にしました．

マイコンで印字制御を行なってみて一番不便に感じた事はハンマ・マグネット励磁タイミングの取り方です．回転する活字ドラムに該当文字が現われた事をプログラムが判断すると，符合する桁のハンマ・マグネットに対し励磁の信号を送ります．

出力ポートはラッチされたままですので，適当なタイミングで励磁を解除してやらねばすべての活字を重ね打ちしてしまいます．このタイミングをプログラム・ループにより作成していたのですが，ある都合によりMYCOM-4のクロックスピードを早めたら，ループタイミングまで変化してしまい，今までのプログラムでは印字ができなくなってしまったのです．

こんな時インターバルタイマーがあれば確実な制御ができるのだがと，次のマイコンの青写真を頭に画いていたわけです．

インターバルタイマーを使うメリットは確実なタイミング制御の他にもあるのです．アドレス・バスの表示が行なえるコンピュータの場合，今プログラムがどのあたりを実行しているか見当がつきます．注意して見てるとわかりますが，印字動作のうちほとんどの時間をタイミング・ループで消費しています．

インターバルタイマーがあればタイミング作りをインターバルタイマーに任せておき時刻到来までＣＰＵに別の仕事をやらせる事ができるのです。

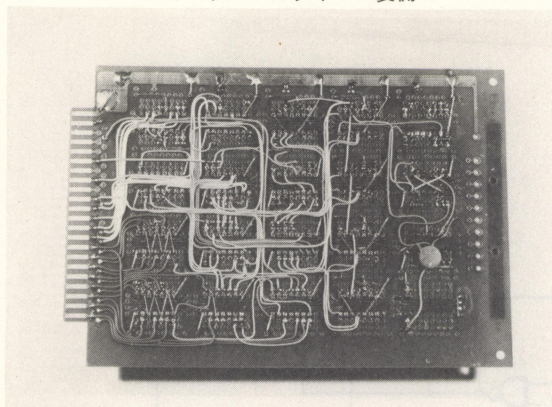
なおこれらの制御が能率よく行なわれるためには，ＣＰＵに割込受付の機能がないとこまるので，ＭＹＣＯＭ-4ではちょっと無理です．

M6800は強力な外部割込を２つ持つため，インターバルタイマーとの連携プレーがより良くできるのです．

●回路の特徴

ここに示すインターバル・タイマーは原型と若干異なり，８種のタイミングしか選べませんが，今のところ別に不自由は感じません．

源信号は１MHzのX' TALからC-MOSにより発振さ

インターバルタイマー裏側

◀前ページ　インターバルタイマー表側

せた後，C-MOSの14518で1/100に分周し，100 μs を作りだします．これをトランジスタを用いてTTLにインターフェイスし，7490ディケード・カウンタ5個を用い，8種のタイミングを作り出します．タイミングの指定はb0～b2を使い，74151のマルチプレクサで選択されます．

タイミングのセット，割込マスクのセット・リセットは8020番地のレジスタを介して行なう事になります（レジスタの機能を表に示しておきます）．

●インターバル・タイマーのトラブル

カードができあがると，前面パネルを使ってHARD・TESTを行ない，タイマー・スタート，タイマー・ストップ，タイミング・セレクト機能をチェックし，すべて良好でした．

さて，プログラムを書き，プログラムでタイマーをスタートさせたりストップさせたりする処理をさせてみました．するとどうでしょう．インターバルタイマーは知らん顔です．シンクロがないので辛苦します．

シングル・ステップで動かすと，ちゃんと動作するので，原因は7474のセット・クロック端子にノイズが飛込んでいるかアドレス・デコード回路の寄生パルスによるヒゲと判断し，CRにより積分を行なう事にしました．はたしてOKとなりました．ここに一つの教訓が生まれました．**『前面パネルでテストしてOKだからといって必ずしもCPUで動くとは限らない！』**

◘総合テスト

第3回目で行なった回路テストにひき続き，今回のテストを行ないます．第3回目と同様CPUカードからはMPUとM6800は外しておく事，代りにICソケットのVccとBAをジャンパーしておきます．

❶スタート・ストップ・テスト

電源投入後に前面パネルのスタートSWを押します．HALTランプが消えるならOK！

次にストップSWを押します．HALTランプが点燈するならOK！（MPUセット時はこのテストを行なわずともよい）

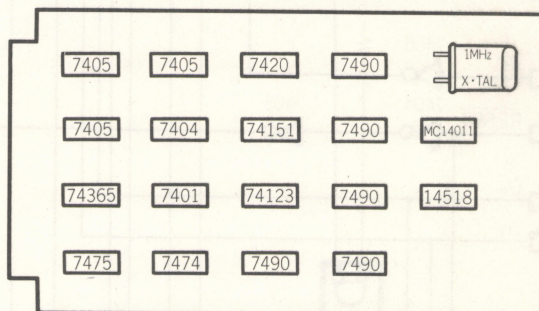
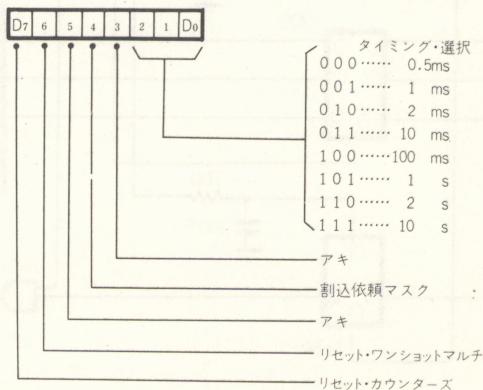
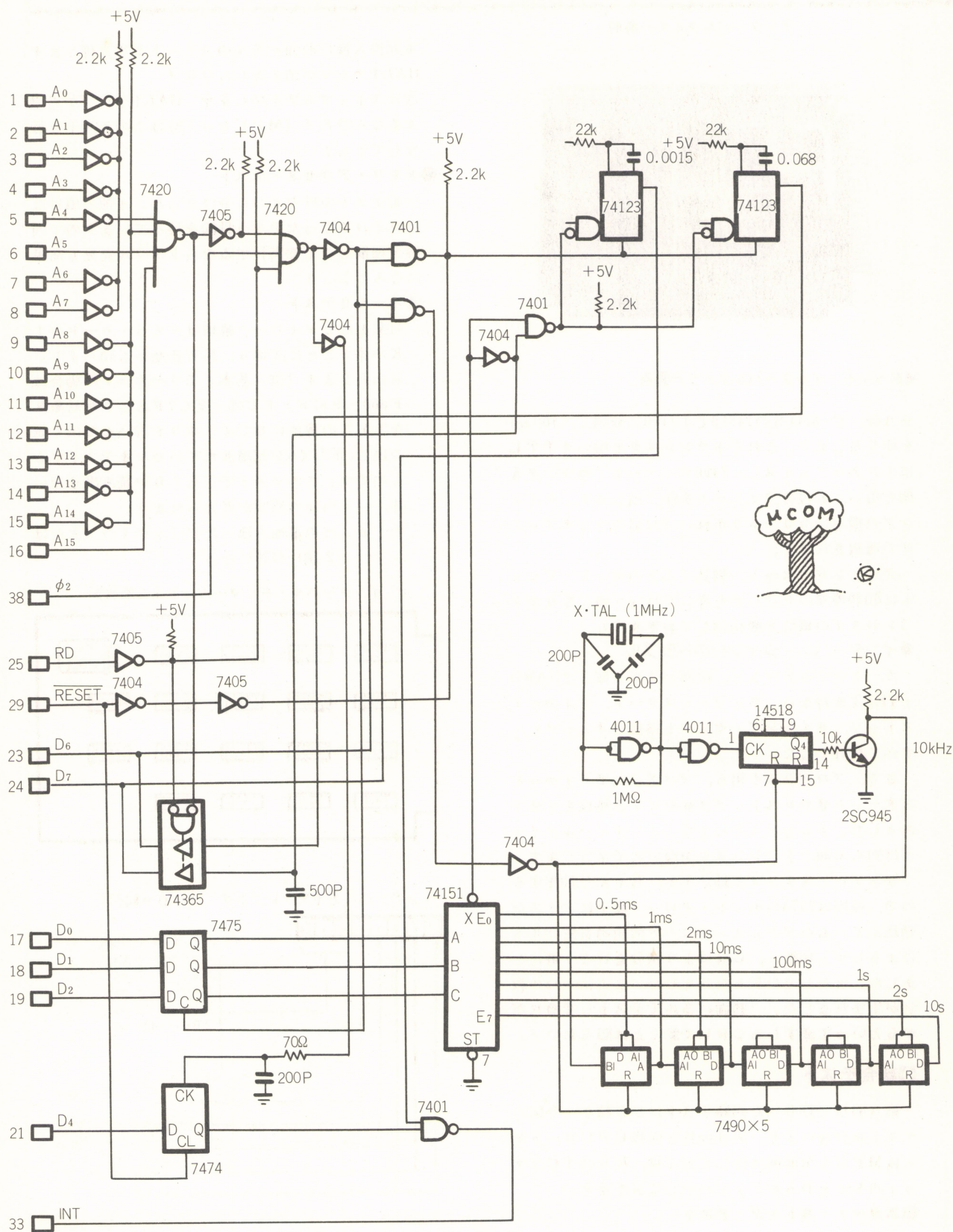
❷メモリ・アクセス・テスト

一応テスト項目を4つに分けました．4種の項目はそれぞれ異なったテスト目的を持っていますので内容を理解した上で実行するならトラブル発見も簡単になるでしょう．

Ⓐオール0テスト

対象となるメモリーの番地はメモリーカードを1Kバイトごとに区切り，最若番地から50バイトぐらいとします（例：基本メモリーカードの場合は，F800～F87FとFC00～FC7Fあたりが対象となる）．この番地に対しオール0を書込んだ後で読み出し，正しく0が記憶されていたかをテストするものです．このテストで正しく0が読み出せない時，いくつかの原因が考えられます．

a．どの番地も決ったビットが1つ（または2個）ONになる

インターバル・タイマーカードIC配置図

インターバルタイマーレジスタ（ADD=8020）

インターバル・タイマーカード回路図

原因＝メモリＩＣのデータ線の配線忘れが多い

b．どの番地もすべてＡＬＬ１になる

原因＝アドレスデコード回路の間違いかメモリーＩＣの入れ忘れ

c．どの番地もランダム・パターンが読み出せる．

原因：メモリスピードが遅いかクロックの速度が早すぎる．あるいはＷＲＩＴ回路の不良で全く書きこまれていない．

Ⓑオール１テスト

対象となるメモリー番地はⒶテストと同じ．

この番地に対しオール１を書込んだ後で読み出し正しく1が記憶されたかをテストします．

a．どの番地もランダム・パターンが読み出せる．

原因：メモリースピードが遅すぎるか，クロックの速度が早すぎる．

b．特定番地だけにビット欠けがおこる．

原因：アドレスデコーダー回路出力にノイズが含まれている．またはメモリーＩＣの不良

Ⓒビット対応テスト

対象となるメモリーの番地は限定しないので，いずれでもよいが256バイトの連続したエリアを必要とする（例：基本メモリーカードの場合 Ｆ800～Ｆ８ＦＦなどがよい）．最初の番地には00を書き次の番地には01という要領で順次＋１した内容を書き込んでゆく．面倒と思われたなら16回でも良い．書きおえたら最初の番地から読み出して内容をチェックする．

a．読出データが80から始まってしまう．

原因：データ表示用ランプの配線ミスかデータＳＷの配線ミス

Ⓓビット・クロストーク・テスト

対象となるメモリー番地はⒶテストと同じ．

このテストの目的は各データ線の信号干渉とメモリーセル内の信号干渉をテストするためであり面倒と思われる人はⒶ～Ⓒが良好なら省略しても良いでしょう．

まず，若番地に対して00，次の番地には01，次の次は02，その次は04，その次は08という具合にまるでネオンサインのごとく順次書き込むのです．これを後で読み出して余分なビットが立っていないか調べるのです．

これらのテストが良好なら，あなたのマイコンは素直に動くと思います．さあ，電源を切ってＣＰＵカードにM6800を正しくセットしてください．セットが終ったら，面倒でも第３回のテストからやり直してください．それが一番早くマイコンを完成させる方法なのです．

◇プログラムによる回路テスト

前項の試験がＯＫならＣＰＵのデータ線とメモリのデータ線がズレていない限り動くでしょう．走行テストに入る前にプログラムの注意と動かし方について説明しておきます．

プログラムを書き込んだなら必ずもう一度読み出して間違いなく書き込まれている事を確認(ベリファイ)してください．プログラムを実行するには，まずプログラム先頭番地をＣＰＵにわかるようにベクトル・アドレスに登録しなければなりません．

一般的なスタートはFFFE番地にプログラムスタート・アドレスの上位アドレスを，FFFF番地に下位アドレスを書き込んでおくだけでよいのです．

あとはRESET・SWを押した後START・SWを押すだけでプログラム実行を開始します．

さて，初めての人には本当に正しくマイコンが動いているのか，全くでたらめに暴走しているのか区別できにくいと思いますのでアドレス・表示ランプの見方を教えましょう．このランプは高速度でアドレス・バス上の信号を表示してます．プログラムの実行というものはコンピュータの性質上アドレスバスの変化率がＬＳＢ(A_0)付近よりＭＳＢ(A_{15}) 付近のほうが緩慢なのです．そこでＬＥＤの光り方もその桁が１になる比率にしたがって光度変化する事になります．

ですからプログラムが整然と実行されている時など，『現在06××あたりのプログラムを実行しているな』と，わかるのです．しかし，暴走を起こしたプログラムの場合は，すべてのアドレスランプが点燈してしまい，星のまたたくがごとき様相となるので区別できます．

Ⓐダイナミック・ストップ・テスト

目的：ＣＰＵがメモリーに格納されたプログラムを正しく読み出して正しい実行を行なっているかをテストします．わずか１ステップのプログラムですが，重要なテストです．

手順：プログラム格納後プログラムをスタートさせます．コントロール表示ランプはＶＭＡとＲＤの２個だけ点燈します．アドレスランプは停止中よりもソフトな光り方でＦＥ01と読める筈です．確認がすんだらストップ・ＳＷを押します．コントロール表示ランプはＨＡＬＴとＲＥＡＤとＢＡが点燈します．ＯＫならもう一度スタートＳＷを押します．これがプログラム実行状態なのです．できたら螢光燈などＯＮ・ＯＦＦしてみてください．暴走が始まったならあなたのマ

テストプログラムⒶ（ダイナミック・ストップテスト）

●ベクター・テーブル

ラベル	アドレス	内容	説明
IRQ	FFF8		
SWI	FFFA		
NMI	FFFC		
RESET	FFFE	FE00	プログラム先頭アドレス登録

●プログラム・リスト

M・CODE		MNEMONIC・CODE			
LOC	INS	LABEL	OP	OPERAND	COMMENT
FE00	20	TEST	BRA	*	DYNAMIC・STOP
1	FE				
2					
3					
4					
5					
6					

テストプログラムⒷ（インタラプト機能テスト）

●ベクター・テーブル

ラベル	アドレス	内容	説明
IRQ	FFF8	FD10	TRAPルーチン登録
SWI	FFFA		
NMI	FFFC		
RESET	FFFE	FD00	STARTアドレス登録

●プログラム・リスト

M・CODE		MNEMONIC・CODE			
LOC	INS	LABEL	OP	OPERAND	COMMENT
FD00	8E	START	LDS	I,X"FFF0"	STACK・AREA ジュンビ
1	FF				
2	F0				
3	0E		CLI	*	
4	3E		WAI	*	
5	20		BRA	*	DYNAMIC・STOP
6	FE				
7					
8					
9					
A					
B					
C					
D					
E					
F					
FD10	01	TRAP	NOP	*	
1	01		NOP	*	
2	3B		RTI	*	
3					
4					
5					
6					

イコンは電源ノイズ対策を行なう必要があります．

Ⓑインタラプト機能テスト

目的：CPUが正しく割込機能を処理できるかをテストします．

手順：プログラム格納後プログラムをスタートさせます．すぐにWAIT状態に入ります．ストップ状態とはちがって，コントロール表示ランプはBAとREADが点燈します．OKならINT・SWを押してください．今度はダイナミックストップ状態になる筈ですからⒶのテストと同じ事になります．もう一度やってみたい人は，このままRESET・SWを押してみて下さい．WAITに入る事ができます．

Ⓒメモリーテスト

テストプログラムⒸ（メモリー・テスト）

●ベクター・テーブル

ラベル	アドレス	内容	説明
IRQ	FFF8		
SWI	FFFA		
NMI	FFFC		
RESET	FFFE	FF10	STARTアドレス登録

●使用上の注意

a．使用者はプログラム・テーブル（FF00番地～FF03番地）にテスト開始番地とテスト終了番地をセットして下さい．

＊例題ではF800～FEFF番地をテストする事になります

b．START・ADD≦STOP・ADDの条件を満足する事

c．自分のプログラムの置かれているエリアをテストしてはならない．

d．テストはスタートアドレスからアドレス増加方向に1番地ずつテストする

e．書込のパターンはオール・ゼロを使用している

f．エラーが発生した場合FF04～FF05番地にエラーとなったメモリー番地が格納されている

g．プログラムの格納場所を変更した際は＊＊印の値を変更しなければならない（アブソリュート・アドレス参照箇所）

目的：マニュアルではすべての番地をテストする事はとうてい無理ですがこれをコンピューター自身にやらせるのです．

手順：プログラム格納後プログラムをスタートさせます．このプログラムはメモリーテストプログラムとしては最も原始的なもので，実用的とはいえませんが，プログラム・ステップ数が少ないので初心者でも扱えます．

機能：開始番地から終了番地まで一番地ずつテストしてゆき，エラーが発生したらエラー発生番地をFD00～FD01番地に登録して割込待ち（HALTの代用）になります．エラーの発生がないときは終了番地テスト後，再び開始番地からテストをくりかえしますので，エラーが起こるまで何日でも動き続けます．筆者は別のプログラムですが，24時間連続テストした事があります．

◘反省事項

ハードウェア製作編は今回で終りです．ここでハードのしめくくりの意味で反省すべき事を掲げておきます．

●製作費について

当初考えていた予算よりも大幅に超過している事に驚いています．最大増設時で25万円（昭和51年11月において）という製作費はアマチュア向であるといえない．しかしパーソナルコンピュータとして仕事をさせるためにはこの程度の能力を持っていないと無理な事

M·CODE			MNEMONIC·CODE			
LOC	INS		LABEL	OP	OPERAND	COMMENT
FF00		F8	STARTADD	DA	2	メモリ・テスト カイシバンチ トウロク
1		00				
2		FE	STOPADD	DA	2	メモリ・テスト オワリバンチ トウロク
3		FF	CURRENT			
4		00	CURRENT	DA	2	テスト・ジッコウチュウ バンチ
5		00				
6		00	DATA	DA	1	エラーノ パターン
7						
8						
9						
A						
B						
C						
D						
E						
F						
FF10		FE	SETUP	LDX	E, STARTADD	
1	**	FF				
2	**	00				
3		86		LDAA	I, 0	BITパターン ケッチイ
4		00				
5		FF	LOOP	STX	E, CURRENT	ゲンザイ ノ テストバンチ オ ヒヨウジスル
6	**	FF				
7	**	04				
8		A7		STAA	X, 0	パターン・カキコミ
9		00				
A		E6		LDAB	X, 0	パターン・ヨミダシ
B		00				
C		11		CBA	*	
D		26		BNE	ERROR	
E		08				
F		BC		CPX	E, STOPADD	1サイクル・テストオワリ？
FF20	**	FF				
1	**	02				
2		27		BEQ	SETUP	YES………モウイチド・ハジメカラ
3		EC				
4		08		INX	*	X·REG＋1
5		20		BRA	LOOP	
6		EE				
7		F7	ERROR	STAB	E, DATA	エラーパターン オ タイヒスル！
8	**	FF				
9	**	06				
A		0E		CLI	*	
B		3E		WAI	*	
C						
D						
E						
F						

も経験上知っていたつもりです．

●ＩＣの数と消費電力

この規模のシステムとしては消費電流が多すぎます．理由もいろいろ考えられますが，その１つは共通バスのインターフェイス用ＩＣが多すぎる事．これはカード面積とも関係があり，大分大きなカードを用いる事で，ある程度は改善されます（特にメモリ基板など有効）

その他に消費電流を減らす方法としてはＴＴＬをＬＳシリーズに変える事で容易にできるのですが製作費の上昇をまねきますので簡単ではありません．またＬＳシリーズは市場供給能力もレギュラーＴＴＬ程になっておりませんから，入手に苦労するでしょう．しかたなくレギュラーとＬＳシリーズ混在型を採用するとしてもファンアウト，ファンインを考慮しなければ回路は動きません．これらの諸問題をふまえて回路は設計されねばなりません．

◆これからのハードウェア

ＣＰＵについてはより高速で機能の優れた物が現われてくると思いますが，同時に周辺ＬＳＩの充実も行なわれるでしょう．M6800についてもＰＩＡ，ＡＣＩＡという周辺ＩＣが存在しなかったら，その秘めたる力を十分に発揮する事はできません．今セカンドソース

の日立からはDMAコントローラ（HD46504)が発表されておりこれらと組合せるならM6800の能力は格段に向上するでしょう．MPUをそのままで周辺LSIを変える事によってマイクロコンピュータの能力が向上するというのは変に思えるかも知れませんが，コンピュータをシステムとしてとらえる目があるなら理解いただける筈です．

これはマイコンに限りませんがLEDの発光効率が数％向上しただけで半導体の世界が変わるような気がします．なぜこんな事を書くかといえば，LEDが消費する電流が決して無視できる量でないし，LED点滅時のノイズ発生がシステム信頼性を下げるからに他なりません．

でばっく・るーむ

8月号P61の回路図およびP63の本文中に誤りがありましたので，ここにおわびして訂正させていただきます．P63下より3行目「一般にICの入力定格はGND≦VIN≦VCCとなって……」と訂正願います

マイコン連盟ニュース

■ミーティング

7月17日のミーティングは，東京秋葉原で行なわれましたが，30名の参加者が楽しくＢＡＳＩＣを勉強しました．全々マイコンを知らなかった人も帰りには立派なプログラムを作って帰りました．
会場にあったF-8使用のＴＶゲームは大変な人気で，高校生 vs 中年紳士の対決など名勝負が続きました．

■APPLE,SWTPC 講習会

９月25日のミーティングでは，今，米国で圧倒的人気のパーソナル・コンピュータ，ＡＰＰＬＥとアマチュアに人気のあるＳＷＴＰＣの講習会を行ないます．

と　き：**９月25日(日)１：00より**
ところ：東京都千代田区外神田１－５－13
昌平橋パーキングビル２Ｆ
（秋葉原駅下車１分）
定　員：30名
会　費：￥1,000(会員，￥2,000(一般)
お申込みは (03)375-5784(工学社) またはハガキでマイコン連盟ミーティング係へご連絡ください．

8080による
マイクロコンピュータの基礎と製作 1

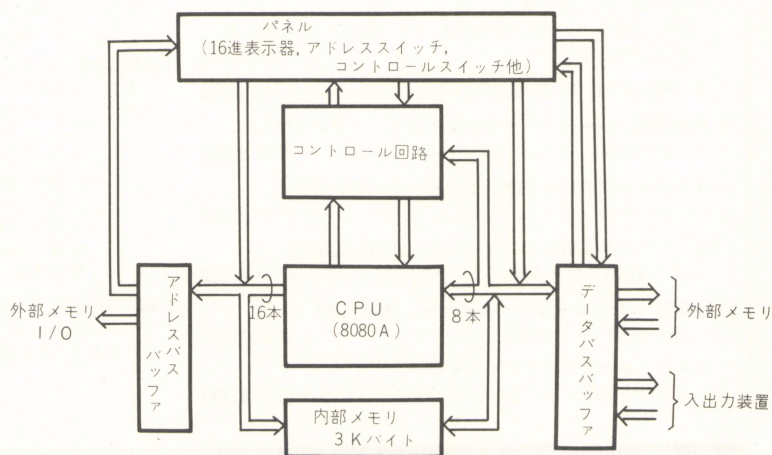
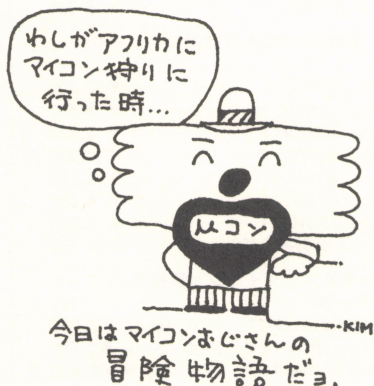
松浦裕之

はじめに

さて今月から数回にわたってマイクロコンの基礎と作り方を紹介してゆきます．すでに6800による荻原氏の連載が５月号から始まっていますが，ここで対象とするＣＰＵは8080にします．8080以外のＣＰＵを使いたいという方にもこの連載が役立つようにしたいと思っています．つまり，基礎的知識や種々のICの事，実際の組み立て方，その他苦労話などをまじえたいと考えています．またすでにマイクロコンを動かしている方には基礎知識など無用でしょうが，それらの苦労話が役立てば幸いです．

安くなる一方のＩＣ

『製作費用はいくらか』とよくたずねられるのですが，正確にはわかりません．というのは，ジャンク(junk＝中古品）や手持ちの部品を多く利用したからで

図1　ブロック図

す．それらを計算に入れても，キットを買うよりは，ずっと安くできたと思います．私が製作に取りかかったのは約１年前ですが，当時ＣＰＵ（8080Ａ）と１Ｋバイトのメモリをあわせて18,000円で購入しました．もちろん新品です！　今ではその半分の値段になっていますね．少し前までは４万円，発売当時はもっと高かったでしょう．

同様の事は電卓にも言えます．数年前は＋－×÷だけの機能のものでも１万円を超えていたのに，今では関数付のものが最低５千円からあるようです．世の中物価値上げがどうのこうのと言われていますが，半導体部品の世界では，逆にはげしく値下りするものが非常に多いのです．と言っても値段が下るのを待っていては話が始まらないわけで，思いきることが必要ですが……

構成について

次に大まかな構成について紹介しましょう．ＣＰＵを中心として，３Ｋバイトの内部メモリ，バスのバッファ，コントロール回路などで構成しています．（図１）

8080用の周辺回路用の専用ＩＣも売られていて，ＴＫ-80などではそれを用いていますが，パーツ代だけを考えるとＴＴＬ　ＩＣを用いて構成した方が安いようです．

マイクロコンピュータの記事を始めて読まれる方のために，くどいようですが言葉を説明しておきます．まず今まで何回もでてきたＣＰＵというのは，Central Processing Unitの頭文字を取ったものです．日本語では中央情報処理装置となりますが，なんのことはない**写真１**のような40ピンのＩＣです．

これによって，たし算ひき算その他の演算や，メモリとデータのやりとり，外部との入出力その他を行なうわけです．どういう動作をするかは，メモリにしまってある命令（インストラクション）によって次々に決められます．したがってメモリは命令を組み合わせたプログラムと，データとが入っているわけです．

メモリは多数の記憶単位からなっていますが，その中から１つを選ぶためにアドレスバス（16本の線）が設けられています．またデータバス（８本）というのは，その名の通りデータが出入りするだけでなくメモリからの命令もここを通ります．8080などのＭＯＳ-ＩＣはあまり多くのＩＣを直接ドライブできないのでバッファをつけるのです．

さて，これだけではＣＰＵはうまく動けません．ＣＰＵがコントロール回路に，『現在どういう状態にあるのか』『これからどういうことをするのか』ということを知らせて，各部の制御を行なわなければなりません．さらに人間とのコミュニケーションのためにパネルのスイッチや表示器がついているわけです．パネルも出入力装置の１つと考えて，プログラムによってその制御をすることも可能ですが，ちょっとめんどうになります．

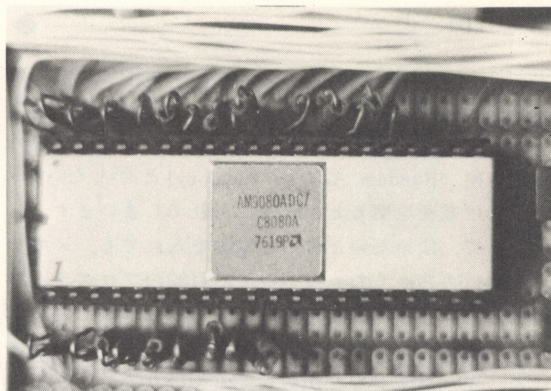



写真１　ＣＰＵの上面

汎用機と専用機

私の制作したコンピュータは，いわば汎用機です．すなわちメモリにはどんなプログラムでも入れることができ，外部との接続部（インターフェイスという）に適当な回路を組めば，いろいろなことができます．パネルにはいろいろなスイッチ群があり，これによってプログラムを開発したり，確かめたり，修正したりできるのです．汎用機に対して専用機または単能機というのがあり，それはある１つの目的だけに用いられるのです.たとえば，電子レジスタの中に組み込んで，レジの制御・計算のみをさせるようなものです．

マイクロコンピュータそのものを趣味にする方は，汎用機をつくるべきですが，マイコンという便利なＩＣを利用してある仕事だけをさせたいという場合には，専用機にしてしま左た方が良いでしょう．そういう時にはプログラムを変える必要がないので固定化してしまい，パネルのスイッチ群や表示器も目的以外のものは不要でしょう．

ただしプログラムを自力で作ろうとするとパネルの機能は必要で，知人に汎用機を借りるということになってしまいます．目的に応じたプログラムが固定化され，ＩＣチップの形で売っていれば良いのですが，いろいろ問題があるようで実現しません.たとえば,アマチュア無線をやっている人のために，モールス信号解読用や，通信の記録整理用のプログラムを，ＩＣメモリのチップに固定化して売っていて，マイクロコンのチップと適当に組み合わせてやれば使える，ということになればいいと思うのですが……　最近はプログラム用の大量のメモリがひとつのチップにはいっているマイコンも出現していますから，それが安くなれば実現

するかもしれません.

メモリのこと

ところで上に述べたような, 内容が変化しないＩＣメモリというのはＲＯＭ (Read Only Memory) と言います. これに対して, 簡単に書き変えができるものをＲＡＭ (Random Access Memory)と言い, たいていのものは電源を切ると内容はこわれてしまいます. したがって, 使いたいと思って電源を入れても, メモリの内容はめちゃくちゃなわけで, 目的のプログラムを毎度入れてやらねばなりません. これは単調な仕事ですが, 長いプログラムでは何時間もかかるかもしれません. キーボードなどがあってもやはりめんどうで, また人間が操作すると入れまちがいも多いのです.

この対策としては,消費電力が非常に少ないC-MOSという種類のＩＣを用いて, 他の部分の電源を切っても, メモリの電源は乾電池で常に供給しておく方法があります. また, オーディオ用のカセットテープを利用して, プログラムを保存する方法もあります. つまり, メモリの内容の０, １の値を適当な回路で音声帯域の信号に変え, カセットテープにとっておくわけです. 逆にそれを再生して適当な回路を通してやれば, 前のプログラムが再現できるわけです.

今, 適当な回路と書きましたが, その回路も普通はプログラムで制御されるわけで, 最低そのプログラムは毎回手で入れてやるか, もしくはＲＯＭに入れておく必要があります.その他, 紙テープリーダー/ライターがあれば利用できます. またコアメモリは電源を切っても, 内容はこわれず, かつ書き替えが自由にできますが, メモリの周辺回路が大変めんどうになります. 不揮発注性ＲＡＭというのもありますが, まだあまり広まっていません. ともかく, プログラムの保存法として, アマチュアの間に一般化しているのはオーディオカセットであると言えます. 私もそれを利用しています.

8080CPUチップについて

さて8080ＣＰＵチップの説明に入りますから図２を見てください. これはＩＣを上から見たものです. まず先ほど説明したアドレスバスが16本, データバスが８本あります. 図２で$_0$～A_{15}, D_0～D_{17}というものです. ここでA_0, D_0というのは最下位のビットでＬＳＢ (Least Significant Bit)と呼びます. またA_{15}, D_7は, 最上位のビットでＭＳＢ (Most Significant Bit) になります. アドレスバスは, 信号がＣＰＵから出ていくのみですが, データバスは信号を出すだけでなく, 入力することもします. こういうのを双方向性バスと言います. つまり, データバスは, ＣＰＵ側からドライブすることも, 周辺回路側からドライブすることもできるわけです.

図２ 8080端子図(上面図)

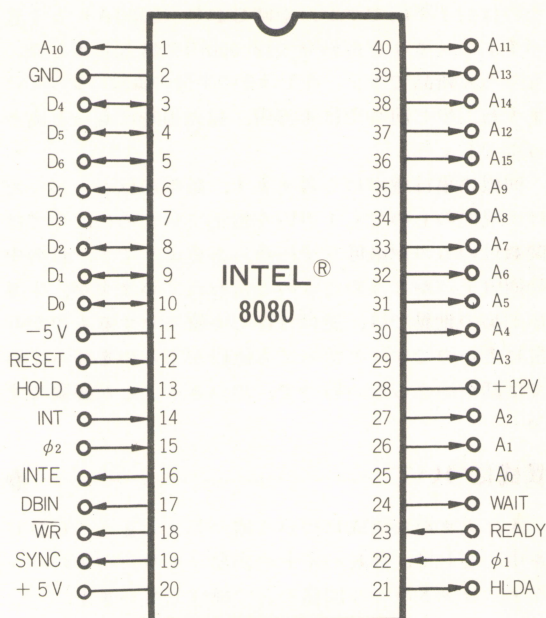



表１ 10進, ２進16進の対応

10進数	２進数	16進数
0	0	0
1	1	1
2	10	2
3	11	3
4	110	4
5	111	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10
⋮	⋮	⋮

注意しなければいけないのは, 両方から同時にドライブしようとするとＩＣがこわれる恐れがあることです. これは一般的に言えることです. すなわち, 今からよく使うTTL·IC (トーテムポール出力) でも忘れてはならないことです. 両方のＩＣが同じ電圧でドライブすればよいのですが, 値が違うと両者の間にけんかが起き, 大電流が流れてＩＣがこわれてしまうわけです. そこで適当なコントロール回路を用いて, ＣＰＵが入力状態の時のみ, 周辺回路がドライブするようにしなければいけません.

図２で15ピン (ϕ_2) と22ピン (ϕ_1) には, クロックを加えます. ＣＰＵはこれによって内部動作の基本的

タイミングを作るわけです．8080Aでは最高2.083MHzまでの周波数が許されています．私は最初テスターのみで作ったので，余裕をもって920kHz にしていましたが，現在は最高周波数で使っています．

RESETというのは，CPUがどんなことをしていても，新たに最初の番地に戻り，その命令から実行をさせるための端子です．番地（Adress）というのは，多数あるメモリのそれぞれの住所のようなものです．普通８進または16進数で呼びます．

16進数について

16進数について一応説明しましょう．10進数なら０から９までの10個の数字を使うし，２進数なら０と１の２つの数字を使いますが，16進数では０～９までの数字の他にA～Fまでのアルファベットを用いて，16個の文字をそろえます．**表１**にそれぞれの対応を示しました．

16というのは２の４乗で，アドレス・バス（16ビット）やデータ・バス（８ビット）を，それぞれ４つ２つに分けると４ビットずつになり，16進数４桁または２桁に相当するわけです．いちいち２進数で言っていたのでは長くなるので，16進数で表わすわけです．たとえば，1110 1010 0011 $1000_{(2)}$は16進数でEA$38_{(16)}$という具合です．右下に$_{(2)}$とか$_{(16)}$とか書いたのは２進数16進数ということを示すためです．

16進数はコンピュータでは，必らず使うのでぜひ覚えてください．先程の16個の１，０の数字を一度に覚えられる人はいいですが，私達凡人には無理です．話は少しそれますが，電話をする時番号を全部覚えるのは無理で，たいていの人は局番をまわしたあとメモをまた見て回すそうです．電々公社では，利用者がどんな回し方をするかを統計的に調べていて，より良いシステムの設計に役立てているとのことです．番号をまわすための時間が長ければ，それに応じた交換機の数にしないと，一ぺんに多くの人が電話しようとしても，交換機にさえなかなか接続されない，ということになってしまいます．

HOLD端子

さて図２に戻って，HOLD端子はCPUを一時停止させ，バスを周辺回路がドライブできるようにする要求の入力端子です．CPUがその要求を受けつけると，HLDA端子がハイレベルになります．これは，パネル面からメモリに直接データを書きこむ時に，操作するとよいでしょう．

READY端子は，やはりCPUの一時停止ですが，HOLDとは止り方が少し違います．これはスピードの遅いメモリなどを用いる時に，ほんの少しCPUを待たせるのに使います．またこの端子でパネルから一時

停止をさせて，現在のアドレスやデータバスの状態を表示させたりもします．

INT端子は割り込み要求，INTE端子は割り込みが可能であることを示す線です．SYNCは，コントロール回路のタイミングを取るのに用います．これらについては後で説明します．

DBINはデータ・バスが入力モードであることを示し，ハイレベルの時に，メモリや周辺回路がCPUをドライブすることになります．

WAITは，READY信号が受け付けられた時か，または『停止しろ』というプログラムに書かれた命令を受け取った時にハイレベルになります．これでパネル面のランプをつければ，CPUが動いているのかどうかがわかります．

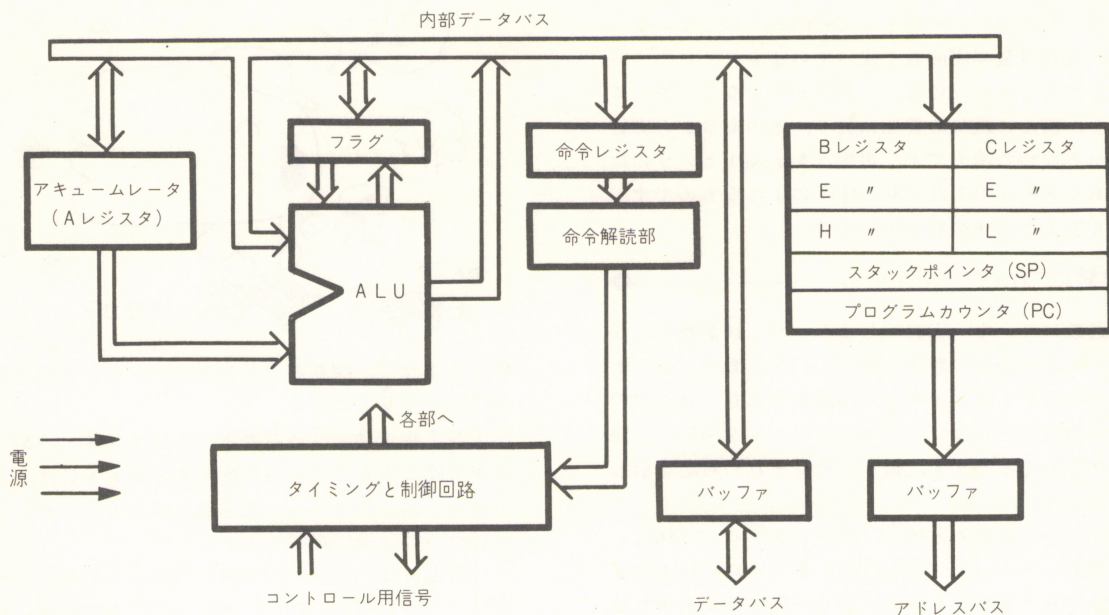
$\overline{WR}$というのは，**ローレベル**になった時CPUのデータバスが出力モードということを示しています．WRという文字の上に横棒（バー）がついていますがこれは負論理を示します．**WR**ite outputという意味の信号がひっくり返っている（ハイレベルとローレベルが入れかわっている）ことを示しています．

さらに電源３種とアースラインがあります．6800やZ80では電源は１種（５V）のみですが，8080ではその他に＋12V，－５Vが必要で，若干めんどうです．

CPU内部のしくみ

さて以上がCPUの外観ですが，その内部はどうなっているのでしょう．それを図３に示します．これは多少省略してありますが，実際に使うには充分です．

データをたくわえるために，アキュームレータ（Aレジスタ）とBCDEHLレジスタの７つのレジスタがあります．データをいちいちメモリからとり出すのはめんどうだから，CPU内に貯えてあるわけです．いずれも８ビット長ですが，BC，DE，HLはそれぞれ組み合わせて16ビット長のレジスタとしても使えます．８ビットでは2^8＝256個の数しか表わせませんが，16ビットなら2^{16}＝65,536個の数まで表わせるわけで便利です．またアドレス・バスは16ビットなのでその制御に

図3　8080ブロック図

も都合が良いのです．

電卓のこと

それでは，『さらに大きな数をあつかうにはどうするのか？』『小数点のついた数は？』『負の数は？』などと疑問を持たれるかもしれませんね．

その問題はソフトウェアをうまく作ってやれば，なんとか解決できます．電卓では10桁程度の四則演算を簡単にやってしまいますが，それは大変なことです．

普通，電卓では10進数1桁毎に4ビットの2進数をあてはめる（BCDコードという）ので10桁なら40ビットになり，さらに小数点や符号のことを考えると長くなります．その上，sin, cos, tan, log, ln, expというようなものもボタンを押すだけで計算可能で，マイコンにそれができるようにプログラムしようとすると，大変なことになります．『電卓はエライ！負けそう！』という言葉がとび出してしまいますが，結局大量生産の結果安いのであって，マイコンの方が実はエライのです．電卓は計算だけが目的ですが，私達がつくろうとするマイクロコンは，他にもゲームとか制御とかいろいろなことができる汎用機なのですから……

ALUのこと

話を図3に戻しましょう．ALUというのはArithmetic Logical Unit）の略で，日本語にすると論理演算素子といういかめしいことになります．が要するに，たし算，ひき算，AND，ORなどという演算をするところです．『かけ算や割算がないではないか？』という方があるかもしれませんが，5，6月号にミスターX氏が書かれているように，それはプログラムで計算させるしか手がありません．ハードウェアでかけ算の回路をつくるのは案外めんどうで，8080にはそのような機能はありません．それは私達がかけ算をする場合，1桁どおしならともかく，大きな数どおしではめんどうになることと同じです．

ALUの上についているフラグというのは，演算の結果が零になったとか，桁あふれしたとかを示すもので，全部で5つあります．

PCのこと

次にプログラムカウンタ（PC）を説明しましょうか．CPUはメモリから次々と命令をとり出して実行するわけですが，めちゃくちゃにとり出すのでなく，PCが示す番地のメモリから命令をもってきて，命令レジスタに入れ何をすべきか解読して，その実行と同時にPCの値はふえ次の番地を指し示すことになります．もしJump命令であれば，その命令に書いてある情報がPCに入り，次に命令をとり出す時には，その番地にとぶ（Jumpする）わけです．

SPのこと

残ったスタックポインタ（SP）というのは，少々やっかいです．スタック（Stack）を英和辞典でひいてみると，『干し草の山』と書いてあります．コンピュータで〝スタック〟という場合には，情報を山のように積んでおくところと考えて良いでしょう．8080では主

としてアドレスの情報，またはデータを積みあげます．そして最後に積んだものが，おろすときには最初にでてくるわけです．これをlast in−first out といったりします．干草の山から最初に積んだものを取り出そうとしても，下の方にあるから無理で，最後に積んだ上のものからしか取り出せないわけです．

さて，スタックはそのような情報を貯えるものですから，メモリと同じことになります．それならば，今までの命令（プログラム）やデータが入っているメモリと同一の物とすることができるわけで，普通そうします．ただし，最後にどこの番地にスタックとして情報を入れたのか覚えておくために，スタックポインタ（ＳＰ）というものがあるわけです．ＳＰの値は情報を読んだり書いたりするごとに増減します．

注意すべきことは，ＰＣは普通１ずつふえるだけですが，ＳＰは情報を書くごとにへってゆくことです．これは図４のようにメモリを使うからです．なお，スタックに情報を書き込みすぎると，命令やデータをこわしてしまうことがあるので，注意が必要です．

苦労話

私は，主な回路を**写真２**のように一枚の基板にまとめてしまいました．この大きさは200㎜×300㎜で22ピンの両面コネクタ部が２つついています．一枚の基板に載せることによって，基板どおしの配線とかコネクタのトラブルなどは，考えなくてよいことにしました．ガラスエポキシ製なので，ちゃんと買えば相当なものです．ちゃんと買わないならどうしたのだ，ということになりますが，実はジャンク基板なのです．これは40個程度のＩＣ（ＴＴＬ，ＤＴＬ）が載っていて 800円，という物でした．何と安いこと！ ２枚買ったのですが，もっと買っておくべきだったと反省しています．一年前に日本橋（大阪）のジャンク屋で見つけたのですが，すぐに売れたようです．裏面からしか，ハンダ付けしてなかったので，比較的簡単にＩＣをはずせました．簡単と言っても，ＩＣをあまり熱くしないように注意して，根気のいる仕事でしたが……．

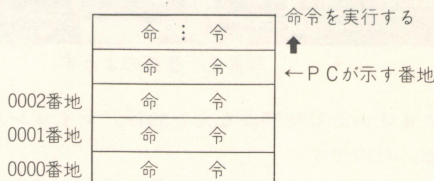
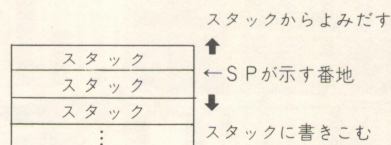



図４　PCとSPの動き方

		スタックからよみだす ⬆
	スタック	
	スタック	←ＳＰが示す番地
	スタック	⬇
	⋮	スタックに書きこむ

	命　⋮　令	命令を実行する ⬆
	命　　令	←ＰＣが示す番地
0002番地	命　　令	
0001番地	命　　令	
0000番地	命　　令	

さらにＩＣをはずしたあとに，ハンダでふさがっている穴をあけるのも，大変めんどうでした．これはあけたい部分を熱しながら，つまようじでつつくという方法でやりました．

皆さんにおすすめしたいのは，『ジャンク屋でユニバーサル基板を見つけたら，即買え』ということです．もちろん，暇が充分ある方に対してだけのおすすめですが……　しかし，こんなことを書くと値段が上るかな．ジャンク屋の皆さん，ＩＣをはずすのは非常にめんどうなのだから，どうか値上げしないで下さい！

ある計算をマイクロコンにやらせたら，36秒かかりました．ところが，某大型コンピュータにフォートランで書いた同じ計算をやらせたら，コンパイル・リン

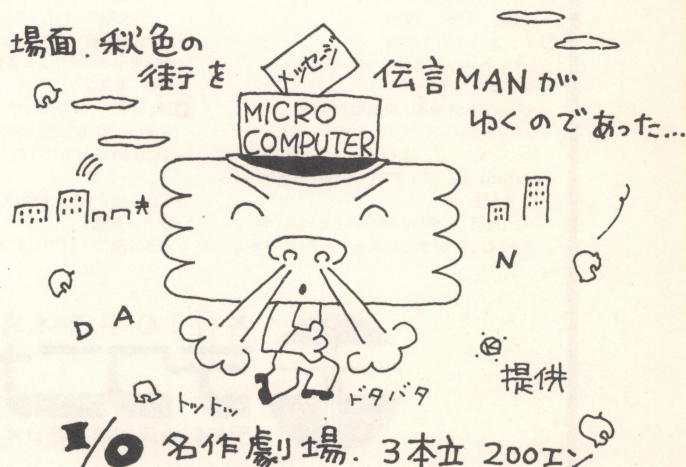
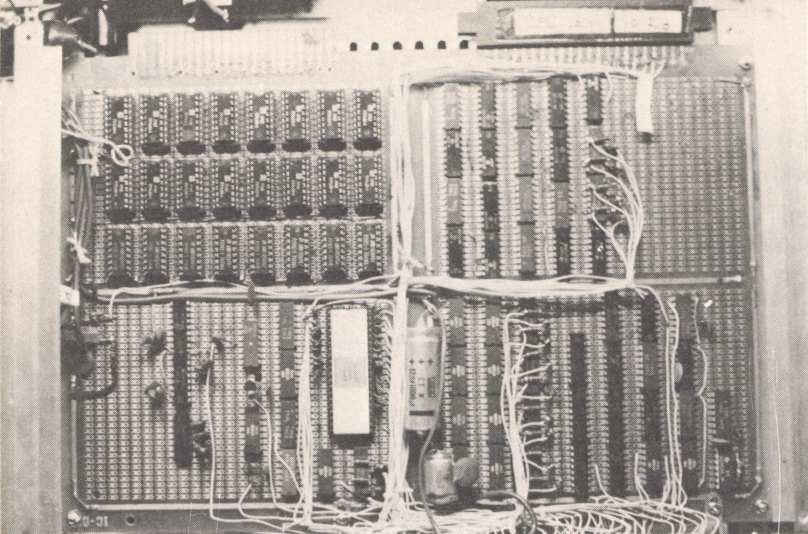
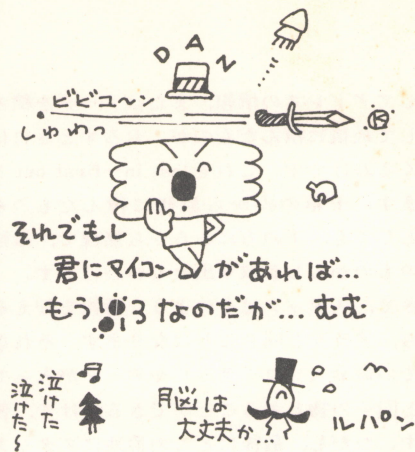



写真2　基板のようす

クの後の正味の計算時間がなんと39秒，マイコンとほぼ同じだったのです．

もちろん，私の書いたフォートランのプログラムが適切でなかったことも大型コンピュータが負けた原因です．その大型コンピュータのプログラムを適当なアセンブラ（機械語）で書いたら，大きな違いがでたでしょうし，出力の方法，その他いろいろな条件の悪さがあったのでしょう．ともかく，こんなことも起こり得るのです．マイクロコンとして，ばかにできません．場合によっては，大型コンピュータよりもすごいことができるかもしれません．

さあ，みなさんがんばりましょう！

◘参考資料

8080　Micro Computer Systems
User's Manual（intel, Sep. 1975）

New Product

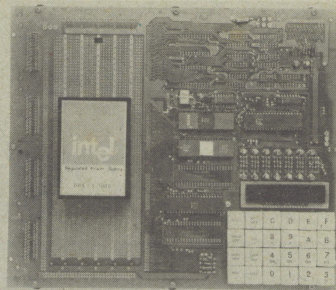
§新CPU8085のキーボード付きマイクロコンピュータキット§

■SDK-85は8080Aを機能強化し，使いやすいものにしたCPU8085とそのファミリ素子MCS-85の完全組立キット

《特徴》

▶CPU8085と新しいファミリ素子8155，8355，8279の4個のLSIで完全なシステム(CPU，RAM，ROM，I/O)を構成，組立てが容易．完全にスタンドアロンのコンピュータである．

▶コンパクトな中に次の機能が入っている．

〔2Kバイトのプログラムメモリ（4Kバイトまでそのまま増設可能)／256バイトのデータメモリ(512バイトまでそのまま増設可能)／38本のI/Oライン(76本までそのまま増設可能)／14ビットのタイマ／5レベルの割込み（キットでは3レベルを使用）／キーボード，LED，TTYインターフェイス内蔵〕

▶24個のキーボード，6桁のLEDを内蔵．

▶システム全体が5V単一で動作．

《価格》　￥81,000

《問い合せ先》　インテル・ジャパン(株)

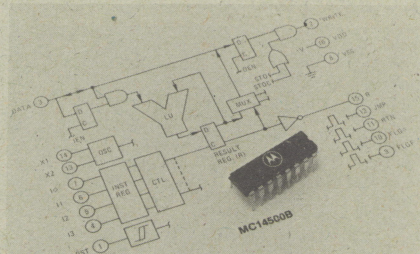
〒154 東京都世田谷区新町1-23-9フラワーヒル新町東館　☎(03) 426-9261

§インダストリアルコントロール・ユニット§

■MC14500B　ICUは，プログラマブル・ロジック・コントローラの中心部をICにしたもの．リレーシステムのようなシンプルさとマイコンのようなプログラムの容易さを目的とし，工業用各種制御に適している．

《特徴》

▶16命令▶手書のプログラム容易▶C-MOSのため，高雑音余裕度，バッテリーバックアップ容易▶スタティック動作▶幅広いクロック周波数など．

《価格》￥2,280(100個以上)

《問い合せ先》

モトローラ・セミコンダクターズ・ジャパン(株)

〒150 東京都渋谷区神宮前6-12-18

☎(03) 499-1231

§8080用フォートランコンパイラー§

■FORT/80は，ANSI FORTRAN IV(JIS FORTRAN 7000レベル相当）のサブセット．マイクロコンピュータ特有の機能を拡張させたコストパーフォマンスの高い8080マイクロコンピュータ応用システムのプログラム開発ツール．

《特徴》

▶14Kバイト以上のメモリがあれば，マイクロ・コンピュータ自身でFORTRAN言語で書かれたプログラムをコンパイルできる．

▶ランタイムパッケージは必要とせず，処理速度の速い即実行型のオブジェクトプログラムを作り出す．

▶オブジェクトプログラムは，直接PROMに書き込める8080の機械語．

▶オブジェクトプログラムの処理速度は，現在あるマイクロコンピュータ用高級言語のどれのものよりも高速．

▶マイクロコンピュータ応用システムのソフトウェア開発，デバッギング，文書化のコストを削減．ソフトウェアの保守改善が容易になる．

▶1バイト，2バイトの整数形演算及び単精度，倍精度のIBM型式の浮動小数点形演算ができる．

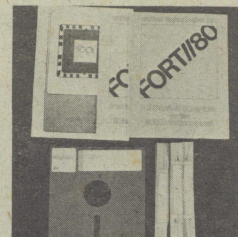
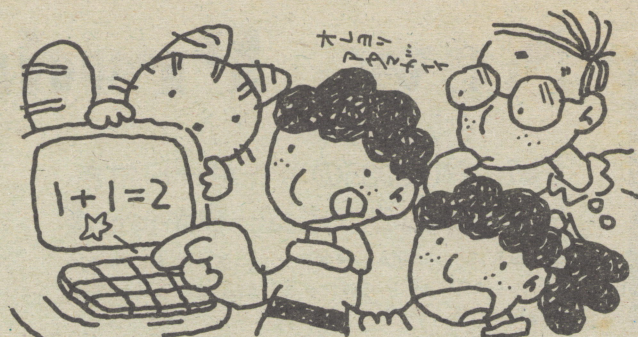
《価格》

1	FORT／／80コンパイラ	498,000円
2	媒体（デイスケットまたは紙テープ）	18,000円
3	FORT／／80言語仕様	3,000円
4	FORT／／80取扱説明書	3,000円

《問い合せ先》

(株)オートメーション・システム・リサーチ

〒105 東京都港区西新橋3-15-8　西新橋中央ビル

☎(03) 437-5471

特訓講座 BASICで遊ぼう！《PART 2》

数に強くなるには

手塚佐知
(コンピュータ・ラブ)

先月はPRINTという命令で文字列をいろいろな形式で印字する方法を説明しました．今回は数の取扱いをマスターしましょう．

1. LETできめちゃおう

```
:10  LET  A=25      Aの値をきめる
:20  PRINT  A
:30  END     ここで終り  "A"でないことに注
                          意！
:RUN          実行開始

 25  ←──  Aの値
:
```

行番号10に新しい命令LETが入っています．これは"Aを25という値にしなさい"ということで，これ以後何の変更もなければ，Aはずっと25という値になっています．ですから次のPRINT命令で，Aの値を印字するわけです．本当かな？ためしてみましょう．

```
:PRINT  A  ←行番号がないのですぐ実行！
 25
:
```

ということで，何度でもAの内容をよび出すことができます．ここで使っているPRINT命令では，Aという文字に" "がついていません．これはAという文字を印字するのではなく，その内容を印字するからなのです．違いをためしてみることにします．

```
:PRINT  "A"
 A
:PRINT  A
 25
:
```

2. LETで計算しよう

```
:10  LET  A=50  ) A，Bの値をきめる
:20  LET  B=25  )
:30  LET  C=A+B )
:40  LET  D=A-B ) 計算してC～Fに答
:50  LET  E=A*B ) を入れる
:60  LET  F=A/B )
:70  PRINT  "A"，"B"，"+"，"-"，"*"，
     "/"  ←項目の印字
:80  PRINT  A，B，C，D，E，F  ←データ
            の印字
:90  END
:RUN

 A     B     +     -     *      /
 50    25    75    25    1250   2
```

LETの後にいろいろと式を書けば，それに従って計算され，結果が指示したところ（C～F）にしまわれます．ここで＊は掛算，／は割算の意味です．

```
LET  Q=((A+B)*C)/(A-B)
```

などということもできますからいろいろとやってみて下さい．

今回は70のPRINT命令は文字の印字，80ではデータの印字が行なわれるようになっています．よくその違いをみて下さい．

3. PRINT命令でも計算できる！

```
:PRINT  5+8
 13
:
```

いかがですか．5＋8と印字されるのではなく，計算されて答が印字されるでしょう．つまりPRINT命令の中に数式の表現ができ，その計算結果が印字されることになります．

```
10  LET   A＝50  ) A，Bを決める
20  LET   B＝25  )
30  PRINT   A＋B      計算結果を印字する
：RUN
75  ←A＋Bの結果
：
```

そこで，2でやった例題をもっと簡単に下のようにすることができるようになります．

```
：10  LET   A＝50
：20  LET   B＝25
：30  PRINT "A", "B", "+", "－", "*", "/"
：40  PRINT  A, B, A+B, A－B, A*B, A／B
：50  END
```

このようにPRINTの中で表現をしますと，ステップ数が少なくてよいでしょう．

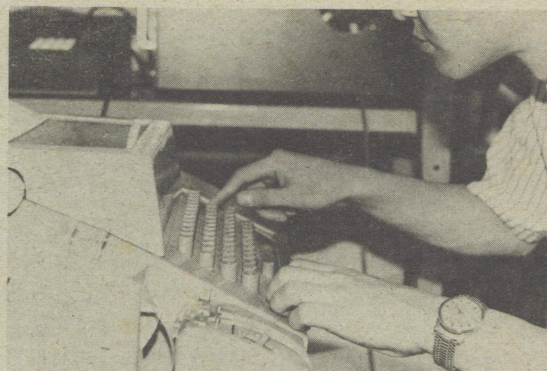
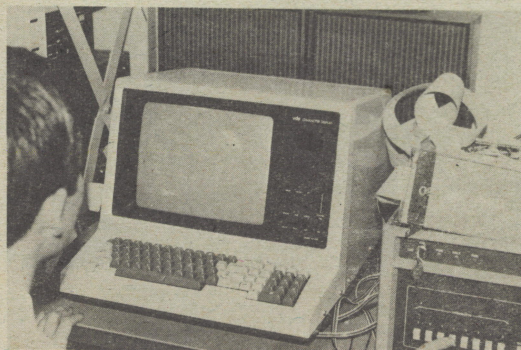
4. データを変えるにはINPUTで

上のプログラムでAやBをいろいろ変えて電卓のように使いたい場合はどうしたらよいでしょうか．まだ前のプログラムが入っていますからやってみましょう．

```
：10  LET   A＝85  ) 行番号10,20を変更する．
：20  LET   B＝10  )
：LIST       ←プログラムのリストをとってみる．

 10  LET   A＝85
 20  LET   B＝10
 30  PRINT  "A", "B", "+", "*", "/"
 40  PRINT  A, B, A+B, A－B, A*B, A／B
 50  END
```

たしかに変更されています．これでRUNとやれば答が得られるのですが，毎回書き直すのは面倒です．

そこで次のようにINPUTという命令を使ってみます．

INPUTとは"データを入れるぞ"ということになります．

```
：10  INPUT  A
：20  INPUT  B
：LIST

 10  INPUT  A
 20  INPUT  B
 30  PRINT  "A", "B", "－", "*", "/"
 40  PRINT  A, B, A+B, A－B, A*B, A／B
 50  END
```

ここで実行させるためにRUNとしますと？が印字されて来ます．これは"入力待ち"ということですから，Aにあてはめる数をキーボードから入れてCRを押しますとまた？がでます．今度はBの値を何にするかきいているわけですから，Bにあてはまる数を同じようにして入れCRを押します．

```
：RUN

 ? 40
 ? 20
 A     B     +     －     *     /
 40    20    60    20    800   2

：
```

というわけで，何度でも自由にデータを入れることができるようになりました．

```
：10  INPUT  A, B
```

とすると，1行で入力が済むことになります．この時行番号20は消す必要がありますので

```
：20  CR
```

とやります．

5. 無条件でジャンプはGOTO

今RUNのかわりに下のようにGOTO 10とやってみます．

```
：GOTO　10　[CR]　←行番号10より実行せよ

　?
```

すると？がでてきます．これは行番号10へジャンプしろということを命令されたので，行番号10より命令が実行されはじめたのです．つまりRUNと同じことになります．RUNは〝行番号の一番小さいものから実行せよということなのですが，GOTOは指定した行番号から，プログラムを実行させることになるわけです．

もしGOTO 30とすれば，すぐPRINT命令に入るため，印字が開始されます．

次にGOTO命令を現在のプログラムに加えてみましょう．

```
：45　GOTO　10
：LIST
　10　INPUT　A，B
　　　　　　20は前に消しました
　30　PRINT　"A"，"B"，"+"，"−"，"*"，"/"
　40　PRINT　A+B，A−B，A*B，A/B

　45　GOTO　10　←新しくつけ加えたもの
　50　END

：
```

こうすると，プログラムは10からはじまり，45まで進んで10に戻ることになりますから，くりかえし四則演算をやるようになります．ちょうど電卓のようになったわけです．

```
：　RUN

　?10
　? 5
　A　　B　　+　　−　　*　　/
　10　　5　　15　　5　　50　　2
　?20
　?10
　A　　B　　+　　−　　*　　/
　20　　10　　30　　10　　200　　2
　?
```

ところで，これでは別の仕事をやらせようとしても，ENDまで行ってくれないので，どうしようもありません．このプログラムから抜け出すには，タイプライタのBREAK KEYかCTRL（コントロール） Kを押す必要があります．それでもだめなら，システムリセットを押さなければなりません．

6. 結局どうすればよいのでしょう？

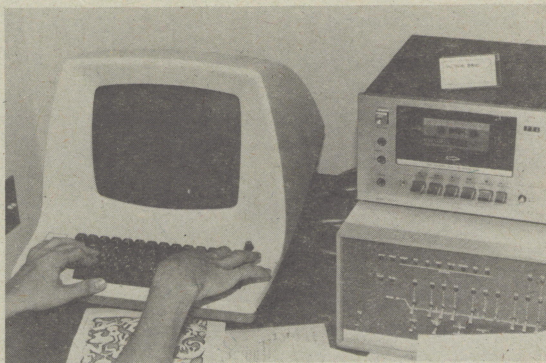
これまでいろいろやって来た命令はどうも一方向ばかりしか進みませんから，魔法使いの弟子ではないですけれど，水をくむ命令だけを与えて，そこらじゅう水びたしという有様になるのです．

〝水が一杯になったら止める〟ということも考えなければなりません．つまりある条件になったら，それまでの仕事から別の仕事に移るということで，これが判断といわれるものになります．

次回はこの判断からはじめましょう．

```
：45　INPUT　Q
：46　IF　Q>O　THEN　GOTO　10
：LIST
```

わかりますか？

中学生の作ったサイコロトバク・プログラム

```
LIST

10 PRINT "**********SAIKORO TOBAKU**********"
20 PRINT "DONATASAN MO YOGOZANSUKA? YOGOZANSUNE?"
30 PRINT
40 PRINT " "," ","HAIRI MASU !"
50 PRINT " ","KARA KARA KARA      PON!"
60 PRINT
70 PRINT " HATCHA IKENAI OYAJI NO ATAMA, HARANAKYA KUENAI"
80 PRINT " CHOCHINYA ARAMA CHOI CHOI YUDEAZUKI NANKINMAME"
90 PRINT " NO TSUNAWATARI! SA HATTA HATTA !!"
93 PRINT "CHO WA (0) DE HAN WA (1) DAYO! "
95 PRINT
100 PRINT " ","A= ",
110 INPUT A
115 PRINT " ","B= ",
120 INPUT B
130 PRINT " ","C= ",
135 INPUT C
140 PRINT " ","D= ",
150 INPUT D
160 Q=RND(6)+1
165 R=RND(6)+1
166 PRINT
167 PRINT
168 PRINT " ",Q,R,
170 S=Q+R-((Q+R)/2)*2
171 PRINT S
175 IF S=0 GOTO 400
180 PRINT " "," ",Q;" ";R;" NO HAN!"
190 GOTO 410
400 PRINT " "," ",Q;" ";R;" NO CHO!"
410 PRINT
420 PRINT " "," ",
421 A=A-S
422 B=B-S
423 C=C-S
424 D=D-S
425 IF A=0 PRINT "A",
426 IF B=0 PRINT "B",
427 IF C=0 PRINT "C",
428 IF D=0 PRINT "D",
429 PRINT "OMEDETO GOZAIMASU!"
430 PRINT
435 PRINT "KOREDE YAMEMASUKA?(1) MO ICHIDO YARIMASUKA?(0)
440 INPUT H
450 IF H=1 GOTO 470
460 GOTO 20
470 END
```

3 4 5 6 7 8 9

S-100 BUSを斬る!—

間違いだらけのバス選び

最近“S-100BUS”というシステムバスが，話題になっています．これは，インテルの8080をCPUとするアメリカMITS社のALTAIR8800シリーズの880-110システムバスのことで，このバスが，事実上アメリカの標準バスと考えられるほど普及しており，日本でも検討を始めたということだと思われます．最近の雑誌の記事などを読むと，S-100 BUSを，検討するというよりも，『S-100 BUSには，太刀打ちできないので，日本でも，S-100 BUSを標準として，進むべきてある』というような，安易な受けとり方をしているように感じられます．そこで，私なりに検討した結果，疑問に思われたことについて、記したいと思います．

◆おかしなバスのピン配列

バスのピン配列を図1に示します．
この図から，分かるように，ピンの配列があまりにも無秩序で，この原因は，CPUのプリント基板作成時に，他のユニット及び，新しいユニットが作成される場合の事を重視せず．CPC基板内のみで，プリントパターン引き出しが楽なようにということだけで決定された，ピン配列ではないかと思われます．

このことは，逆に，VI_0～VI_7までの，8本のベクトル割込線が，他のバス・ラインに比して，あまりに秩序よく配列列されていることと，この8本の割込ラインは，割込ラインをジャンパーで，決定しているため，プリントパターン上では，他の信号とは異なり，接栓部のみに，ジャンパー用ラウンドを用意しておけば，内部素子の，配列には，無関係に，信号線を配置することができ，このために配置を考える場合，客観的に，秩序を持った配置で，ピン番号を決定できるということで，このように，整然と配置されたと思います．

このように考えると，『機能を重視して，ピン配列をしたというより，プリントパターンの引き出しが楽なようにピンを配列したのではないか？』ということも考えられます．（ただし，信号線はできるだけ短かくということで，見かけ上の秩序を無視したということであれば，私の誤解ですが.）

◆他のピンにも変なのが……

次に，実際不都合と思われるピンの配置について考えてみましょう．

ⓐピンNo.49のCLOCKとNo.99のPOC，この2つの信号のクロストークがシステムとしての大きな信頼度に関係するであろうと思われます．

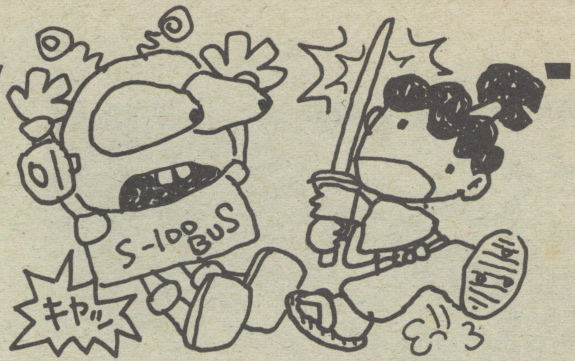
それは，POCとは電源投入時のリセット信号で，CPU外のI/Oなどのユニットにほとんどに入力され，初期状態の保持を目的とする信号であり，この信号と平行にCLOCKの信号が，あるということは，クロックの信号が，POCに，誘導されたりするしシステム全体が誤動作する原因とも考えられます．

ⓑピンNo.50および100のGNDバス方式のシステムで，最も重要な信号の基準レベルとなるGND（0V）用接栓が部品面及びハンダ面に1ピンずつ用意されているだけで．ALTAIR, IMSAIなど，20スロット以上ユニットを実装可能というシステムのバスにしては，貧弱すぎるのではないでしょうか．実際には編組線などを使用して，補強していますが，補強は最終的な手段で最良ではありえません．

ⓒピンNo.2および52の+16V,−16V．このピンも，GND同様システムバスにしては，貧弱であり，この2本のピンは接栓が向い合っており，このためバスボード内でのパターンの流れが，コネクタのピン間を通ることになり，必然的に，細くしなくては他のピンとショートすることになるため，導体幅も小さくなってしまいます．このためラインドロップ．インピーダンスなども大きくなると考えられます．

また，基板実装前の電圧チェック時などはピンが向い合っているために，+16V,−16V間をショートさせてしまうという事故を，起しかねません．

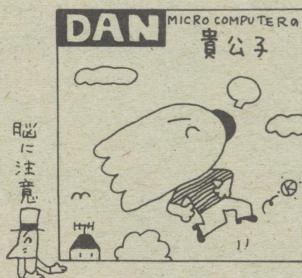
このように最悪でも，部品面，パターン面を共通の電源として使用できるようにすべきであると思います．

ⓓピンNo. 36～38，D_0，A_{10}．このピン配列は，専門外の方が見ても，分かると思われるほど，ピン引き出しが，いいかげんである．他のピンは，ある程度配列に対し理由付けはできると思いますが，〝A10″と言う，アドレスバスが，DO0及びDO4のデータ出

力バスの間に引かれているというようなことは，他のユニットを製作する場合にもユニット内での素子配列ミスや，PIN－No の読み違いの原因ともなり，またDOの信号とＡ１０のＣＰＵより送出されるタイミングの違いによるクロストークの心配も出て来ると思います.（ただしクロストークは他の信号線の総てに言えるので，特に記しません.）

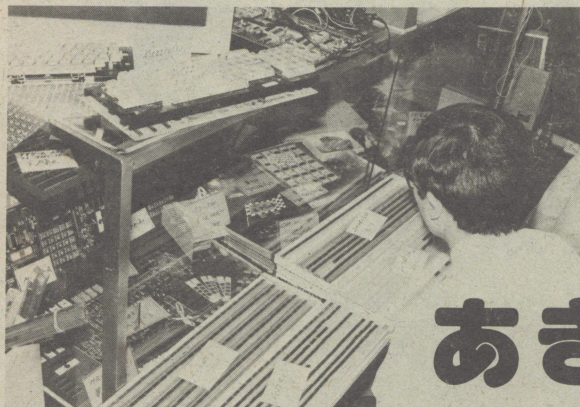
ⓔロジック電源とグランド
（基板両サイド）.

ピン配列表を見ると分かりますが，ピンNo.50，１００にグランド．１，51に＋８Ｖとあるように，この条件でプリント回路を作成すると，ほとんどの場合，電源ラインが，回路を囲んでしまうという，構成になってしまい，ノイズ点で考えてると基板外にはノイズのシールド効果が考えられるかもしれませんが，電源ループがループ内の回路に与える影響の方が大きいのではないでしょうか．このことは，電源が安定化されて，供給されていれば良いのですが，Ｓ-１００ ＢＵＳでは，ローカル，レギュレータ方式を取っているためにリップルを含んだ電源が基板内を，囲んでしまうということが起こります．この対処の方法は＋電源に，できるだけ近い所にレギュレータを置くようにし，グランド回路でループするようにします.

●以上ピンの配列についてだけでもこれだけの疑問点が上がります.

このように考えると，私はＳ-100は最悪に近いバスだと思えてなりません．良い対処の方法がありましたら，御指示ください　（N）

TOP OFBOADD　　BOTTOM OFF BOARD

PIN.NO			PIN.NO		
1	＋８Ｖ	*	51	＋８Ｖ	*
2	＋16Ｖ	*	52	－16Ｖ	*
3	ＸＲＤＹ		53	ＳＳＷ－ＤＩＳ	
4	ＶＩ０		54	ＥＸＴＣＬＲ	
5	〃 １		55		
6	〃 ２		56		
7	〃 ３		57		
8	〃 ４		58		
9	〃 ５		59		
10	〃 ６		60		
11	〃 ７		61		
12			62		
13			63		
14			64		
15			65		
16			66		
17			67		
18	３ＴＡＴＵＳ－ＤＩＳ		68	ＭＷＲＩＴＥ	
19	Ｃ/Ｃ　ＤＩＳ		69	ＰＳ	
20	ＵＮ ＰＲＵＴＥＣＴ		70	ＰＲＯＴＥＣＴ	
21	ＳＳ		71	ＲＵＮ	
22	ＡＤＲ－ＤＩＳ		72	ＰＲＤＹ	
23	ＤＯ－ＤＩＳ		73	ＰＩＮＴ	
24	０２		74	ＰＨＯＬＤ	
25	０１		75	ＰＲＥＳＥＴ	
26	ＰＨＬＤＡ		76	ＰＳＹＮＣ	
27	ＰＷＡＩＴ		77	ＰＷＲ	
28	ＰＩＮＴＥ		78	ＰＤＢＩＮ	
29	Ａ５		79	Ａ０	
30	Ａ４		80	Ａ１	
31	Ａ３		81	Ａ２	
32	Ａ１５		82	Ａ６	
33	Ａ１２		83	Ａ７	
34	Ａ９		84	Ａ８	
35	Ｄ０１	*	85	Ａ１３	
36	Ｄ００	*	86	Ａ１４	
37	Ａ１０	*	87	Ａ１１	
38	Ｄ０４	*	88	Ｄ０２	
39	Ｄ０５		89	Ｄ０３	
40	Ｄ０６		90	Ｄ０７	
41	ＤＩ２		91	ＤＩ４	
42	ＤＩ３		92	ＤＩ５	
43	ＤＩ７		93	ＤＩ６	
44	ＳＭ１		94	ＤＩ１	
45	ＳＯＵＴ		95	ＤＩ０	
46	ＳＩＮＰ		96	ＳＩＮＴＡ	
47	ＳＭＥＭＲ		97	ＳＷＯ	
48	ＳＨＬＴＡ		98	ＳＳＴＡＣＫ	
49	ＣＬＯＣＫ	*	99	ＰＯＣ	*
50	ＧＮＤ	*	100	ＧＮＤ	*

あきはばら地図（マップ）

あきはばら すぽっと No.11

《Mr.デバラマンの新秋葉原事情》

＊関東バイトショップに，MINIBUGのROMが売られている．型番はSCM44506Pで２万５千円．

＊若松の広告を見ると，日立のCRTコントローラが売られているが，一般向けの小売りは年末ごろになるとのこと．ＳＭＣのものが１万９千円ぐらいであるそうだ．

＊コンピュータ・ラブ

日本一の，のんびりショップ．社長が趣味でやっているような店だ．あんまり，のんびりしているので，まわりの人が心配してくれるそうだ．しかし，我々にとっては気楽でよい．客が多い割りには，売り上げが少ないので，コンピュータ・ラブをつぶさないためにも売り上げ向上に，みなさん協力してあげましょう！　それからなりふりかまわずネバッテ，帰りにはBASICが自由に使えるようになろう！

ここでは，中学生もBASICを使うぞ～．

マイコン用電源

＊秋葉原エレクトリックで，コンピュータ用電源ユニットが２Ｋで売られています．

容量３Ａ，電圧はいろいろあるようですが６Ｖのものを買うと良い．（電圧は30％程度可変できる．）

買うときには，親か子（親に子をつけることによって容量を６Ａにすることができる）に注意！

クローバがついていないので，安全のためにリモートセンスの配線を抵抗で結んでおくとよいでしょう．（図１参照）

あまり数がないようなので，早い者勝ち！

図１　Tr技76－2

バスドライバー

＊若松で，８Ｔ96，８Ｔ97コンパチのものが350円で売ってます．他の店の半値ですが，それでも高いという人は，**ロビン電子**で，Ｔ４Ｌ9366を195円で買ってきて使えばよいでしょう．

７４ＬＳ365→	８イ95
366→	96
367→	97
368→	98

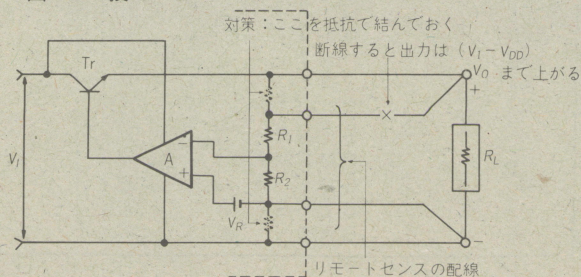
というふうにおきかえができます．

＊ＬＳ・ＴＴＬはロビン電子が安い．たとえば，

７４ＬＳ75	￥175
ＬＳ365	￥195
ＬＳ74	￥135

秋葉原ひろしといえども，これだけ安いところはない．

＊亜土電子は，駅からはなれているが，歩くだけのことはある．たとえば，

２Ｎ3055	￥250
Ｄ5101－Ｅ	￥2800
Ｓ1998（ＡＭＩ）ＭＭ5316同等	￥800
ＬＥＤ10ヶ	￥150から
ＩＣソケット　JAE　14Ｐ	￥500
16Ｐ	￥550

KEL　14P￥450
　　　16P￥500

　ICソケットは，TEXASのものが使い易い．似たものでTHMのものが，安くでているが，秋葉原では見かけない．

＊**アスターインターナショナル**で，近々，キーボードを２種売り出すそうだ．１万７千円程度のものと，ホールICを使ったものだそうだ．(ホールICのキーボードは，４万円以下で出したいとのこと．)

　CPU内蔵のCRTディスプレイも予定しているそうで，乞御期待！

＊**ついに，モールス符号解読用LSI登場！**

　よく見ると，TMS6011．D店に行くとショーケースの中にある．

＊最近，μPD503（８×51220M）を，店頭で良くみかけるが，いったい何が入っているやら．誰か調べて見ませんか？

＊富士無線のとなりの**丸三電機**では，コネクター，ツマミ，放熱器などを扱っている．

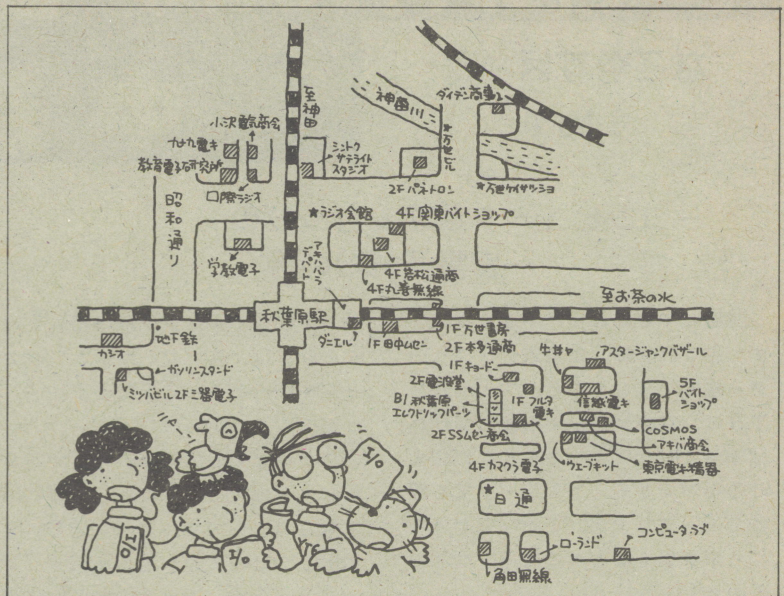
　中でも，放熱器のはかり売りは安い！

＊丸三電機の向い，**WAVEKIT**では，シンセサイザの特売をやっている．興味のある人は，一度，行ってみるとよい．ここではM8071が3,800円，セットで6,200円です．

＊TO－3型の３端子レギュレータは，600円ぐらいするが，**亜土**では，LM309Kが，たったの400円！

＊マイコンは，あまり値引きしてくれないが，日通ビルの西隣り，明芳ビル２階の**アブコ・エレクトリコ**では，6800DⅡが７万で売られていた．測定器の中古品がたくさんあるが客は少ない．

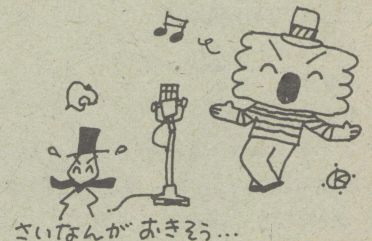
＊近道，ラジオデパートの１階のフルタ電機の横をぬけると，吉野屋の前に出る．そのまま，吉野屋のビルの中に入って行き，エレベータの横のドアから外にでて，右に行くと，信越のところに出る．(アナタとわたしだけのヒ・ミ・ツ……)

＊牛丼屋の通りを信越とは逆方向，

つまり中村電気パーツセンターのほうに行くと，右手に**オートスナック**がある．ここは〝対話のあるオートスナック〟なのだそうだ．メニューは，ボンカレー300円，マックバーガー130円他にもいろいろある．なかでも，おもしろいのは，マックバーガーの販売器で，残り時間を表示する．気が向いたらどうぞ．

＊牛丼屋の上の，**音キチクラブ**には，娘さんらしい女の人が，アルバイト（？）している．I/Oの読者の観賞にも十分耐えられると思う．しかしあまり，本性をむきだしにしないこと．アイスティー 250円なり．マンガ本を読むなり，将棋をさすなり，２～３時間はネバろう．

＊**トヨムラ**では，TOKOのパーツを扱っている．**東ラジ店**ではジャンクも扱っているので，注意するとよい．

＊トランスを捜すなら，**東京電気精機**に行くとよい．オヤジの話では，うちのトランスは表示されている値の２倍はとれるとのこと．もちろん，秋葉原規格での話．他にも測定器のジャンクやパーツがいっぱいあるが，ここでの目玉は，トランスが安いこと．

＊ラジオデパート三階の降りエレベータのところの南側の店，通路のところにあるパーツに安いものがある．最近，２Pトグルスイッチが100円から120円になった．

＊アスターのジャンクバザーの前を湯島通りまで行くと，**トリオのサービスステーション**がある．マニュアルやパーツなんかも売ってくれるそうだから，無線機のこわれた人は，いって見るとよい．

＊**シントクエコー**　では，月曜から金曜の５時20分から，生中継をやっている．秋葉原で買い物して，そのころになったら，いってみよう．チョット，早目に行くと，ラジオ出演のチャンスもある．そしたら，帰りはオミアゲつきだ．カワイコチャン・スターもくるので，帰りにサインをネダろう．I/Oをもっていって，それに書いてもらうとよい．小生のI/Oなんか清水由貴子のサイン入りだぞ．値ウチが違うンだ．ユッコー！暑さのために気が変になったので，今回はここで終り～．ユッコー！

東京と大阪の間でガンバッているマイコンファンのためのなごやすぽっと No.1 中京地図(マップ)

ーコンピュータ端末ショップの一宮電子(株)の巻ーーー

開店してからもう一年近くなります．主な商品は，光電式テープリーダー端末プリンタなど，フォト式ＰＴＲは，電磁ブレーキを使わなければIKWARDがなんと3秒近くで読ませる事ができる．７Ｋワードでも27秒で入ってしまう．カセットで入れてる事を思えばこの差はキョウイのものでしょう．

一宮電子では，その他1702➡500円
723(AVR用)➡200円
8Ｖ10Ａ
18Ｖ４Ａ } ➡2000円
のようです．

プリンタは，日立製で以前Ｍ銀行で使っていたもの．文字は22文字／秒の早さ．紙送りスキップフィードキャリッジリターンもすべて自動だからとても使いやすい．

英文カナ文字記号と16×８文字分が装備されている．すべてマグネットドライブで2ＳＣ458と，2SC1096をダーリントン接続すればラクに君も使えるだろう．

動作中はかなりやかましいので電源ON，OFFもCPUでやらせるとよい．プリンタは，２万円でＰＴＲは2.5Ｋ円です．これの改造用回路は近く発表するつもりです．

ＰＴＲのテープ読取りは特にデータの存在は，スプロケットというテープ中間の小さい穴で決めると（プログラムで）良いでしょう．

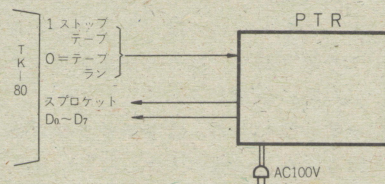
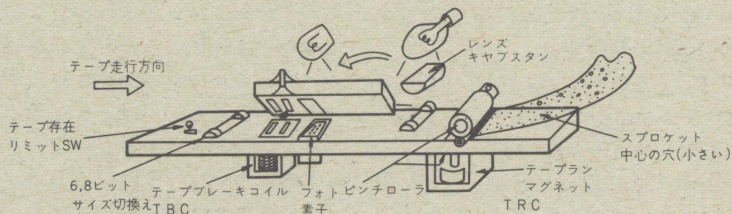
スプロケット信号はテープがあるスピード以上にならないと出力に出て来ないようです．データ信号はその限りではなく光が通る所はず～と０レベルになったまま，データは補数化して出てくるので，A_{cc} で反転する必要があったりなかったり（穴があいてないのが１レベル）

ＴＢＣとＴＲＣは，ＰＴＲの中に制御回路が入っているのでＴＴＬレベルで簡単に動作させる事ができる．

また，電源は，ＡＣ100Ｖ，＋30Ｖ４Ａ，±30Ｖ10mA，＋30Ｖ４Ａは，白熱ランプと２つのコイル，ＴＢＣ，ＴＲＣに使用されＰＴＲのほぼ動力源．

ＰＴＲの中身自身はかなりスペースがあるのでＡＶＲとトランスを組み込んだ方が外からの線数も少なくなり，又ファンが回っているため放熱も良いようです．

±30Ｖは，＋40Ｖ近くで発振させて，±30Ｖを使う事もできますがトランスの21Ｖタップでも良いようです．

センターダプの０Ｖはトランスの線を引きずり出して，中間の線を切って使用して下さい．

ＰＴＲ（フォト式の説明）

（正美）

大須マイコン・ゾーン

東海地区のＩ／Ｏ愛読者諸君．オホン！

日頃，「東京・大阪しか，マイコン・ショップはねえのか!?」とおっしゃっていたことだろう．（「そうだ！そうだ」とバックが声援）おまとうさんでした．今回は我々が，かってに名古屋地区マイコン・ファンを代表して，ただ今注目の大須マイコン・ゾーンを探訪してきたので，ここにレポートするしだいである．オオ！モノホン．

あのねェ．大須というのは東京や大阪でいえば，浅草や道屯堀に当たる所で大須観音を中心として，戦前はそれは華やかであったそうな．フム，フム！そこに今年市営地下鉄三号線（鶴舞線）が開通したことで，再び活気を取りもどしつつあるのだ．

そこで我らマイコン・ファン（自称マコちゃんファンといっとる.）はここぞとばかり，独断と偏見の目でマコを求めて大須マイコン・ゾーンに潜入してみた．

「その態度が気にいらん」ちゅう人は自分で行ってきたまえ、なのだ．

では，いざマコを求めて「マコちゃんヤアーイ！」

ナゴヤパーツ店ガイド

<名古屋マイコン・システム・センター>

この店はマイコン関係のＩＣやＬＳＩそしてシステム・キットの店なんだけど，全般に相談室てな感じが残ってんだなァ～．店員さんが，「マイコンは何をやりたいのかによって選ぶキットが違ってくるから気軽に相談してください．」と言われるとおり各メーカーの品がほとんど取りそろえてあった．

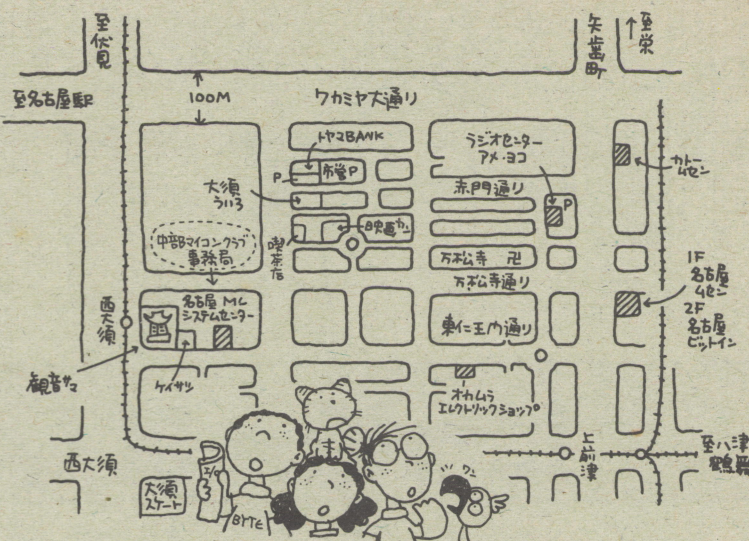
そして，マイコンに興味のある人，さらに今のマイコンを拡張したいと考えている人のために，毎週日曜日に勉強会を開催しているのだそして注目すべきは誰れでも参加できるということである．場所はこのセンターの勉強室．へたな大学の講義室よりイイんだヮ．(学長見てないだろネ？）テーマはソフトとハードを交互に行なわれ，原則的には一週間前に発表される．申し込みは電話なんだト．特殊なテーマは専門家を呼んで開いている．

さらにセンター内に，中部マイクロ・コンピュータ・グラフがあり，クラブでも市民講座などを開催して活発な活動をしている．

さて店員さんが，「メーカーから与えられたものや本だけではわからない事が多い．」と我らの心を見越していわれた．そこで「初歩から開発まで，技術指導を行ないます．」とくるのだが……，店員さんの話はさらに熱気を帯び各メーカーの特徴へ，さらに各種ビットへと続く．この方，マコ・ファン転じて○○○の感じ，何だかたのしくなってきた．

最後に，各社のＰ－ＲＯＭに使用可能なＰ－ＲＯＭライターや，オンライン端末機，高級言語で大型機を使ってやりたい人のための機器が装備されており，申し出れば使用できる．店員さんも「使わにゃソンソン！」といってたョ．使かったるガネ．マットッテチョウョ．

☎ (052)232-1851

<Bit-INN名古屋>

Bit-INNはＮＥＣ専門のマイコン・サービスルームなんだ．ショールーム形式になっていて，製品展示，カタログ，工作・マシンなど６コーナーがあるのだ．マシンコーナーはデバックやシュミレーションができるように設備されており，P-ROMライターなどの無料サービスと指導をしているのだ．

ここではマイコンでのT.V.ゲームや鉄道模型などのデモンストレーションをしていて，このためのインターフェイスの作り方も指導しているので，初心者には具体的でわかりやすいと思うョ．

僕の感じだと，店内がきれいにまとまっていて，落ち着いたショールームという印象だネ．腰を落ち着けたら動くのがイヤになった．T.V.ゲームもあったし……

さらにこちらにもＮＥＣマイコンクラブがあり各行事を行っているので，詳細はBit-INNで聞いて下さい．やさしいお兄さんがマコの世話をしていたョ．

☎ (052)263-0971

<奥村電機>

こちらはコンピュータの端末機専門店なのだ．マイコン・キットはサンプル程度．さらにコンピュータ関係の洋雑誌も売っていた．一見してマコファンのハイクラス向きなのだそれよりもプロ級かな．

この店の特徴はジャンク製品が多く置いてあることだネ．ほとんどのジャンクが市価の10％程度で売っているのだ．たすかるよねェー．端末機は高いのが難点じゃけんネ．店員さんはキーボードやタイプライターへのインターフェイスの自作を勧めており，そのための図面が用意されてるのだ．また製品を買った人のために，その機種を対象とした利用のための講習会を開いている．

☎ (052)264-0005

<ラジオセンター・アメ横ビル>

秋葉原の電機店が共同で名古屋に進出してきたビルなのだ．パーツ屋，オーディオショップ，電機商，ジャンク屋とあるが，マイコン関連店は以下のようなものがある．

◆名古屋Byteショップ

I/O７月号に記事のあった放電プリンタを取り扱っている．プリンタの実演もしていた．各社のマニュアル本があった．

◆タケイ無線

キーボードによるＣＲＴディスプレイの実演をしていた．

<カトー無線，パーツセンター>

名古屋では老舗的なパーツ屋である．広く一般のパーツ屋でほとんど何でもあるもんネ．

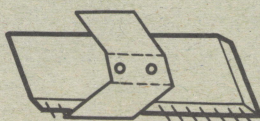
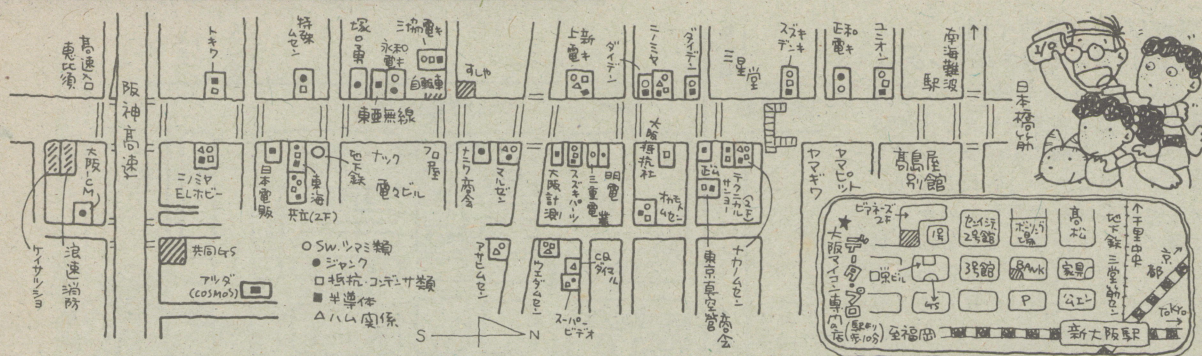
8080,8228用放熱板作成

8080ＣＰＵの金属の所．ここに80W～100Wのハンダゴテを当て，放熱板をハンダ付けする訳です．手早くやらないとＣＰＵは仏になるので注意して下さい．

今までさわれなかった程のＣＰＵが手でさわれるようにはなるようです．もちろんシリコングリスも付けてね．

（正美）

関西マイコンファンのための

にっぽんばし地図(マップ)

★マイコンキットを買ったが，メモリー容量が足りなくて，拡張しようとしている人，または，チップから集めて作っている人にとっては，基板上の配線が，悩みのタネだと思います．（エッ！君は，エッチングしているって？ゴクロウサン！）ビニール線を使うと，コテで溶けたり，基板が，ソバの大盛りになったりしてしまいます．こんな時，便利なのが，ミレン電線なのですが，結構，いい値がします．（**データプロ**で，40円/m，10m単位）できることならこれを使いたい所ですが，我々アマチュアにとっては，少々痛いので，他を探すと，**トキワ**で，ワイアリング・ペンが¥1,950で売っています．これは，ポリウレタン線を，ボビンに巻き，ペン状のケースから，線を引き出すものです．ポリウレタン線は，普通のエナメル線と違い，ハンダゴテで熱すると，絶縁被膜が取れて，ハンダ付けできるというもので，大変便利なのですが，使ってみると線が細い（0.1㎜）ので，基板の表をはわせるのには，少したよりない感じがします．0.3～0.4㎜位が，適当だと思います．これは以前，**ELホビー**で，10m50円で売っていたのですが，今は，ありません．また入るかもしれません．また，電線の事ならここ，と言われている**三重電業**には，300m1,000円であります．困っている人は，一度お試しを．

注意！買う時はエナメル線とまちがわない様に！ポリウレタン線は，キラキラと光沢がある．

ポリエチレン線0.32㎜10m160円ELホビー，コテで溶けない．

μCOM μCOM

日本橋パーツ店ガイド
★ちょっと一言

岡本無線（■○□）☎ 633-5671

前号で紹介したように，マイコン・スペシャル・コーナーを開設したのですが，ＩＣや書籍の数，種類が，もう一つといったところで，このコーナーの気さくな店員の人の話では，ぼちぼち増やしていくという事です．ＭＳ3004￥1,500，ＬＳ365￥215，2102，500nsローパワー￥700など．

また，このコーナーは，シンセサイザーにも手を広げるそうで，ローランドのシステム100をおいています．

◇Ｈ68／ＴＲもおいているので，興味のある人は，ここの店員の人に，聞けば，ていねいに説明してくれます．

◇ビデオＲＡＭ1632は10万弱で，ちょっと手が出ないが出る人はどうぞ．

◇78Ｈ05　￥3,100

共立電子産業（●○■□）☎631-5963

この店は，他の店にくらべ，目先がきくので，繁盛しているようだ．

この間も，インターフェイス・エイジ５月号を店においたらしいが，すぐに売切れ，僕も，くやしい思いをした．ザンネン！

◇Ｚ80vs8080A and 6800（ＭＯＳＴＥＫ）　￥1,000

◇ＭＫ3880（Ｚ80）和文テクニカルマニュアル　￥1,700

◇３端子電源ＩＣ　μPC143
＋５Ｖ，＋８Ｖ，＋12V，＋15V，１Ａ　￥380

◇C-ＭＯＳも案外安い．
ＭＣ14538　￥490

東亜無線（■）☎644-0111

ここにもマイコン・コーナーがあるが，伸光のキットを売っているので，本誌の広告でもおなじみですね．

◇伸光キット、リズムジェネレーター、10リズム、８音　￥13,800

◇16line　タッチセンサー　ＩＣ
ＡＭＩＳ9263　￥4,900

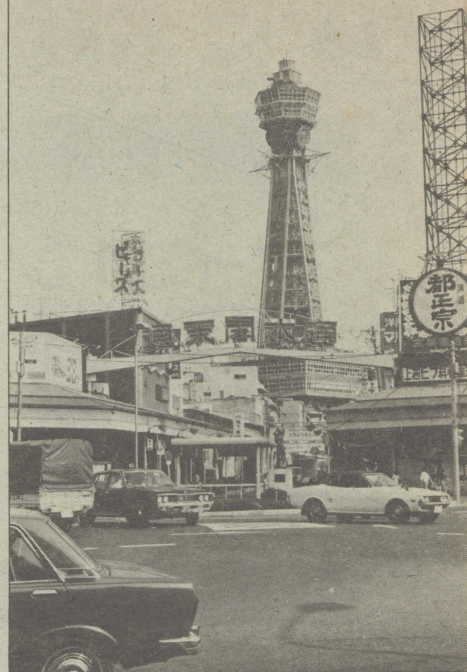
◇ＭＫ3880　￥16,500
ＭＫ3881　￥7,500
ＭＫ3882　￥7,500

★新情報★

■ＨＭ4704-2（250ns）　￥2,000
ＨＭ4711-3（200ns）　￥2,320
ＨＭ435101（650ns）　￥3,200
ＬＭ380（power amp）　￥280
4558　￥240　（日本電販）

■14Ｐ，16Ｐ，ＩＣソケット￥60
24Ｐ　￥120　（大阪ＩＣＭ）

■ＭＣ6800Ｌ　￥6,900
3850（Ｆ－８）￥12,300
3851　￥10,500
ＮＥ556Ａ　￥350
ＮＥ555　￥170
２Ｎ3055　V_{CBO}100V，
I_C15Ａ，P_C115W　￥290
ＣＡ3140　￥440
723　（電源ＩＣ）　￥210
74Ｃ164　￥560
ＭＭ80Ｃ95Ｎ　￥240
ＭＣ7800Ｃシリーズ1.5Ａ　￥460
ＭＣ7900Ｃ　－1.5Ａ　￥550
ＭＣ78Ｍ00Ｃ　0.5Ａ　￥410
ＭＣ79Ｍ00Ｃ　－0.5Ａ　￥450
（トキワ）

■μＰＤ5101（800ns）　￥2,500
μＰＤ2101　￥950
Ｍ54821　ＡＭ／ＳＷ／ＦＭ
周波数表示用ＬＳＩ（I^2L）￥4,300
（テクニカル・サンヨー）

■モニタＴＶ　ケースなし（本誌６月号Ｐ34の写真のＴＶ）　￥7,800
（明電）

■ＴＭＳ1955　￥2,800　（三協）
（IK^2EI）

ボクもひとこと

６月号の「にっぽんばしマップ」で，放電プリンタがかたや14,000円（オカモト）かたや16,000円(共立)と，大枚2,000円も差があったので，早速ひやかしに行った所，すでにどちらも14,000円になっていました，残念（？）

電車賃をソンしたと思いかけた所で，大発見!! プリンタ用インターフェイス基板が16,000円(オカモト)，13,000円(共立)と逆転しているではないか！　早速共立の店員の方にほめてやると，『東京でも13,000円で売っている所があるから安くしました，安い店の板ばさみになって，クルシイ・クルシイ!!』と言っておりました．

正直でたいへんよろしい，しかし！それでも私は買わないのだ，ゴメン。

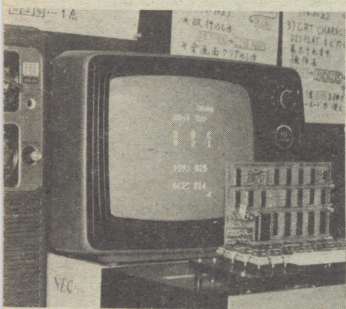
（にっぽんばしムシ）

9月号付録　第２巻第９号　通巻11号　昭和52年９月１日発行(毎月１回１日発行)　昭和52年１月11日第３郵便物認可

○マイコン・キットを床の間に飾らないための

マイコン新聞 BINARY バイナリー

企画編集　東大マイコン同好会

No.2　〔レコード付〕

1.スロットマシン・プログラム
FSK　1200 Hz／2400 Hz　110ボー

2.6800　4K　BASIC
FSK　1200 Hz／2400 Hz　300ボー

3.8080　2K　BASIC
FSK　1200 Hz／2400 Hz　110ボー

スロット・マシン プログラム(8080)

神谷慎悟

大変暑い日が続いていますがマイコンホビーストの皆さんはいかがおすごしですか．今月は，8080システム＋Video・RAMで動かすスロットマシンのプログラムを御紹介します．

キットから始めてBASICを動かすまでには至っていないけれども，Video RAMなら何とかという方に簡単に動かせるVideo RAMの強みとおもしろさを味わっていただきたいと思います．ではプログラムおよびゲームにつて説明していきます．

プログラムとしては，数字を次々と０→９のループでインクリメントしておき，16進キーボードからの入力があるとカウントを停止して表示する方法をとっています．プログラムエリヤは＄8000番地から＄81B1番地までの433 byteを使用しています．Video RAMはFE００番地からFFFF番地までの512 byteを占めています．プログラムのディティールについてはプログラムリストのコメントを参考にして下さい．

なおキー入力ルーチンはＴＫ－80のモニタ内のルーチンを用いていますので，他の8080システムで実行させるときは，別にキー入力ルーチンを作ってやる必要があります．また，もちろんRAM領域や，Video RAMのアドレスデコードが異なるときには，JUMPアドレスを変更する必要がありますので注意して下さい．

さて，プログラムを入れ，8000番地から実行させると，次のような順で処理が行なわれます．

❶ 画面に図１のように出る．
❷ ここで入力キー（リセット以外）を押すと，トクテン－１，カイスウ＋１が行なわれ，図２のような画面が出る．
❸ キーを３回押すとＡから順に数字がはいる（図３）
❹ 以下の規則に従って得点が増える．

Ａ＝Ｂ	＋５
Ｂ＝Ｃ	＋５
Ａ＝Ｂ＝Ｃ	＋30
Ａ，Ｂ，Ｃが偶数のみ	＋１
Ａ，Ｂ，Ｃが奇数のみ	＋１
Ａ＝０	＋１

❺ ここでもう一度キーを押すと再び図２と同じ形式となり，このくり返しとなります．

❻ トクテン＝０になるとENDと出てゲームは終わります．

❼ トクテン＝200になると WDN と出て停止します．

❽ カイスウ＝200になるとOVERとでて停止します．

❾ 停止した段階でRUNを押せばイニシャライズが起こり，図１の画面からスタートします．

◆プログラムの開発およびDEBUGには，NEC のTK－80＋アドテックシステムサイエンスのTVD-02を用いました．なお，資材および資料に関しては，《ムーンベース》の御協力をいただきました．

（なおムーンベースでは，このシステムで実演しています．）

◘参考文献

1）TVD-02 説明書，アドテックシステムサイエンス

2）トランジスタ技術1977年８月号〝TK-80 による…″ CQ出版社

3）つくるマイコンP 97より〝8080の機械語対アセンブラ言語（Intel社資料）″ CQ出版社

✣付録のレコードは，FSK 2400 Hz／1200 Hz の 110ボーで録音されています．

図１	図２	図３
スロット　マシン	スロット　マシン	例
A　　B　　C	A　　B　　C X　　X　　X	A　　B　　C
トクテン　　010 カイスウ　　000	トクテン カイスウ	0　　1　　2

スロットマシンゲーム・プログラムリスト

```
CLEAR 8000 3E 20       MVI A $20        ┐全画面クリア
         2 01 FF FD    LXI B $FDFF      │
         5 03          INX B            │
         6 02          STAX B           │
         7 21 01 00    LXI H $0001      │
         A 09          DAD B            │
         B D2 05 80    JNC $8005        ┘
INIT  800E    21 8C FE LXI H $FE8C      ┐"スロットマシン"
      8011 11 88 81    LXI D $8188      │を出す
         4 06 08       MV1 B $08        │
         6 CD 68 81    CALL $ 8168      ┘
         9 2E CC       MV1 L $CC        ┐"A  B  C"
         B 36 01       MV1 M $01        │を出す
         D 2E DO       MVI L $DO        │
         F 36 02       MVI M $02        │
      8021 2E D4       MVI L $D4        │
         3 36 03       MVI M $03        ┘
         5 21 4C FF    LXI H $FF4C      ┐"トクテン 010"
         8 CD 68 81    CALL $8168       ┘を出す
         B 2E 8C       MVI L $8C        ┐"カイスウ 000"
         D CD 68 81    CALL $8168       ┘を出す
WAIT  8030 CD 16 02    CALL $0216       キー入力を待つサブ
         3 3E 18       MVI A $18        ┐ルーチン
         5 21 EC FE    LXI H $FEEC      │
         8 77          MOV M A          │
         9 2E FO       MVI L $FO        │"XXX"を出す
         B 77          MOV M A          │
         C 2E F4       MVI L $F4        │
         E 77          MOV M A          ┘
         F 21 93 FF    LX IH $FF93      ┐回数を1 ふやす
      8042 7E          MOV A M          (200になるとOVER
         3 FE 39       CPI $39          へ飛ぶ)
         5 CA 4C 80    JZ $804C         │
         8 34          INR M            │
      8049 C3 63 80    JMP $8063        │
         C 36 30       MVI M $30        │
         E 2B          DCX H            │
         F 7E          MOV A M          │
      8050 FE 39       CPI $39          │
         2 CA 59 80    JZ $8059         │
         5 34          INR M            │
         6 C3 63 80    JMP $8063        │
         9 36 30       MVI M $30        │
         B 2B          DCX H            │
         C 34          INR M            │
         D 7E          MOV A M          │
         E FE 32       CPI $32          ┘
      8060 CA 4D 81    JZ $814D         OVER へ飛ぶ
         3 21 53 FF    LXI H $FF53      ┐
         6 7E          MOV A M          │
         7 FE 30       CPI $30          │
         9 CA 70 80    JZ $8070         │得点を1 へらす
```

```
         C 35          DCR M            │
         D C3 81 80    JMP $8081        │
      8070 36 39       MVI M $39        │
         2 2B          DCX H            │
         3 7E          MOV A M          │
         4 FE 30       CPI $30          │
         6 CA 7D 80    JZ $807D         │
         9 35          DCR M            │
         A C3 81 80    JMP $8081        │
         D 36 39       MVI M $39        │
         F 2B          DCX H            │
      8080 35          DCR M            ┘
ABC   8081 CD 75 81    CALL $8175       ┐スイッチを押すた
         4 CA 81 80    JZ $8081         │びにＡＢＣの順に
         7 21 EC FE    LXI H $FEEC      │ランダムな数を入
         A CD 81 81    CALL $8181       │れる
      808D CD 75 81    CALL $8175       │
      8090 CA 8D 80    JZ $808D         │
         3 2E FO       MVI L $FO        │
         5 CD 81 81    CALL $8181       │
         8 CD 75 81    CALL $8175       │
         B CA 98 80    JZ $8098         │
         E 2E F4       MVI L $F4        │
      80A0 CD 81 81    CALL $8181       ┘
COUNT 80A3 21 EC FE    LXI H $FEEC      ┐"A  B  C"
         6 4E          MOV C M          │の内容をCPU内の
         7 2E FO       MVI L $FO        │レジスタに読こむ
         9 56          MOV D M          │(A)→Cレジスタ
         A 2E F4       MVI L $F4        │(B)→D  〃
         C 5E          MOV E M          ┘(C)→E  〃
         D 7A          MOV A D          ┐
         E B9          CMP C            │
         F C2 CO 80    JNZ $80CO        │
      80B2 BB          CMP E            │
         3 C2 BB 80    JNZ $80BB        │
         6 06 1E       MVI B $1E        │(A)＝(B)…5点
         8 C3 CB 80    JMP $80CB        │(B)＝(C)…5点
         B 06 05       MVI B $05        │(A)＝(B)＝(C)…30点
         D C3 CB 80    JMP $80CB        │それ以外…0点
      80C0 BB          CMP E            │をBレジスタに入れる
         1 C2 C9 80    JNZ $80C9        │
         4 06 05       MVI B $05        │
         6 C3 CB, 80   JMP $80CB        │
         9 06 00       MVI B. $OO       ┘
         B 79          MOV A C          ┐(A)＝0ならば
         C FE 30       CPI $30          │Bレジスタに1を加
         E C2 D2 80    JNZ $80D2        ┘える
      80D1 04          INR B'
         2 79          MOV A C          ┐(A)(B)(C)偶数 ┐ならば
         3 E6 01       ANI $O1          │  〃  奇数 ┘Bレジ
      80D5 4F          MOV C A          │スタに1を加える
         6 7A          MOV A D          │
```

```
         7  E6 01     ANI  $01
         9  57        MOV  D A
         A  7B        MOV  A E
         B  E6 01     ANI  $01
         D  5F        MOV  EA
         E  B2        ORA  D
         F  B1        ORA  C
      80E0  C2 E4 80  JNZ  $80E4
         3  04        INR  B
         4  79        MOV  A C
         5  A2        AND  D
         6  A3        AND  E
         7  CA EB 80  JZ   $80EB
         A  04        INR  B
ADD   80EB  78        MOV  A B       Bレジスタの数だ
         C  FE 00     CPI  $00       け得点を増やす.
         E  CA 2A 81  JZ   $812A     ZERO へ飛ぶ
      80F1  05        DCR  B
         2  21 53 FF  LXI  H $FF53
         5  7E        MOV  A M
         6  FE 39     CPI  $39
         8  CA FF 80  JZ   $80FF
         B  34        INR  M
         C  C3 16 81  JMP  $8116     DELAY に飛ぶ
         F  36 30     MVI  M $30
      8101  2B        DCX  H
         2  7E        MOV  A M
         3  FE 39     CPI  $39
         5  C  0C 81  JZ   $810C
         8  34        INR  M
         9  C3 16 81  JMP  $8116     DELAY に飛ぶ
         C  36 30     MVI  M $30
      810E  2B        DCX  H
         F  7E        MOV  A M
      8110  FE 31     CPI  $31
         2  CA 5C 81  JZ   $ 815C    WON へ飛ぶ
         5  34        INR  M
DELAY 8116  0E F0     MVI  C $F0     得点を1ずつ増す
         8  3E 00     MVI  A $00     はやさを決める時
         A  3C        INR  A         間かせぎルーチン
         B  FE 00     CPI  $00       8117のF0を小さ
         D  C2 1A 81  JNZ  $811A     くするとゆっくり
      8120  0C        INR  C         得点があがる.(0
         1  79        MOV  A C       0でもっともゆっ
         2  FE 00     CPI  $00       くりになる.)
         4  C2 18 81  JNZ  $8118
         7  C3 EB 80  JMP  $80EB     ADD へ飛ぶ
ZERO  812A  21 51 FF  LXI  H $FF51   得点が0ならばE
         D  7E        MOV  A M       NDに行き, 0で
         E  FE 30     CPI  $30       なければ WAIT
      8130  C2 30 80  JNZ  $8030     へ飛ぶ
         3  23        INX  H
         4  7E        MOV  A M
         5  FE 30     CPI  $30
         7  C2 30 80  JNZ  $8030
         A  23        INX  H
         B  7E        MOV  A M
         C  FE 30     CPI  $30
         E  C2 30 80  JNZ  $8030
END   8141  36 04     MVI  M $04     "トクテン END"
         3  2B        DCX  H         を出す
```

```
         4  36 0E     MVI  M $0E
         6  2B        DCX  H
         7  36 05     MVI  M $05
         9  76        HLT            停止
         A  C3 00 80  JMP  $8000     CLEAR へ飛ぶ
OVER  814D  06 04     MVI  B $04     "カイスウ OVER"
         F  21 91 FF  LXI  H $FF91   を出す
      8152  11 A0 81  LXI  D $81A0
         5  CD 68 81  CALL $8168
         8  76        HLT            停止
         9  C3 00 80  JMP  $8000     CLEAR へ飛ぶ
WON   815C  36 17     MVI  M $17     "トクテン WON"
         E  23        INX  H         を出す
         F  36 0F     MVI  M $0F
      8161  23        INX  H
         2  36 0E     MVI  M $0E
         4  76        HLT            停止
         5  C3 00 80  JMP  $8000     CLEAR へ飛ぶ
SUB   8168  0E 00     MVI  C $00     INIT用サブルー
         A  1A        LDAX D         チン
         B  77        MOV  MA
         C  13        INX  D
         D  23        INX  H
         E  0C        INR  C
         F  78        MOV  A B
      8170  B9        CMP  C
         1  C2 6A 81  JNZ  $816A
         4  C9        RET
SUB2  8175  78        MOB  AB        ABC 用SUB
         6  3C        INR  A
         7  27        DAA
         8  47        MOV  B A
         9  C5        PUSH B
         A  CD 23 02  CALL $0223
         D  FE FF     CPI  $FF
         F  C1        POP  B
      8180  C9        RET
SUB3  8181  78        MOV  A B       ABC 用SUB
         2  E6 0F     ANI  $0F
      8184  C6 30     ADI  $30
         6  77        MOV  M A
         7  C9        RET
DATA1 8188  7D 5B 6F                 "スロットマシン"
         B  44 20 4F
         E  7C 5D
DATA2 8190  44 78 43                 "トクテン010"
         3  5D 20 30
         6  31 30
DATA3 8198  76 72 7D                 "カイスウ000"
         B  73 20 30
         E  30 30
DATA4 81A0  0F 16 05                 "OVER"
         3  12
AUTO  81A4  21 EC FE  LXI  H $FEEC   XXXに000
         7  36 30     MVI  M $30     を入れて COUNT
         6  2E F0     MVI  $F0       に飛ぶ
         B  36 30     MVI  M $30
         D  2E F4     MVI  L $F4
         F  36 30     MVI  M $30
      81B1  C3 A3 80  JMP  $80A3
```

4K BASIC文法

命令

コマンド	ステイトメント		ファンクション	
LIST&	REM	END	ABS	LEN@
RUN	DIM✻1	GOTO✻	INT	ASC@
NEW	DATA	ON…GOTO✻	RND	SOR@
SAVE	READ	ON…GOSUB✻	SGN	EXP@
LOAD	RESTORE	IF…THEN✻	USER	LOG@
PATCH &	LET✻	INPUT✻	TAB	VAL@
APPEND	FOR	PRINT✻	PEEK	CHRS
DIGITS &@	NEXT	PATCH✻	SIN@	STRS@
LINE &@	STOP	RETURN	COS@	LEFTS@
CONT &@	GOSUB✻	POKE✻@	TAN@	RIGHTS@
PORT&@		DEF@	FNX@	MIDS@
TRACE ON&@			POS@	ATAN@
TRACE OFF&@				

⁑印のステイトメントはダイレクト・モードで使える。
& 印のコマンドは，プログラムの中でも使える。

演算	比較
↑べき乗　@	＝等しい
ー負符号	<> 等しくない
⁑乗算	<より小さい
／除算	> より大きい
＋加算	<＝小さいか等しい
ー減算	> ＝大きいか等しい

@印のものは８ＫＢＡＳＩＣのみ
⁑１印は，４Ｋ ＢＡＳＩＣではダイレクトモードで使えない。
●ライン・ナンバーは１～9999。
●変数はアルファベットとアルファベットの後ろに０～９の整数をつけたもの。
●バック・スペースは〝コントロール０〟
●ライン・キャンセルは〝コントロールＸ〟
●パニック・ボタンは〝コントロールＣ〟これを使うとユーザー・プログラムが何をやっていても READY にもどる。

エラーメッセージ

1．ERROR　　　IN LINE
　Ａ　LINE＃0000は，ダイレクト・モードで起こったエラーを示します。
2．エラーコード
　1．変数オーバーサイズ（255以上）　TAB, CHR, 添字, ON
　2．入力エラー
　3．無効な文字，変数
　4．プリント・リテラルの最後に〞がない
　5．DIM のエラー
　6．無効な演算
　7．行番号がみつからない
　8．ゼロで割算
　9．サブルーチン・ネスティングが８以上
　10．GOSUB なしで RETRURN
　11．間違った変数
　12．解釈できないステートメント
　13．カッコのエラー
　14．メモリ・オーバー
　15．添字エラー
　16．８レベル以上のFOR ループ
　17．対応するＸなしの NEXT　X
　18．FOR－NEXT の対応エラー
　19．READ ステートメント・エラー
　20．ON ステートメントのエラー
　21．入力オーバー・フロー（72字以上）
　22．DEF 文法エラー　@

メモリーマップ

アドレス	内容
0000 – 00FF	インプットバッファ 変数一時記憶域
0100 – 0101	ハード BASIC 開始番地
0103 – 0104	ソフト BASIC 開始番地
0105 – 10FF	BASIC インタプリタ
1100 – 11FF	算術 FOR NEXT スタック
A000 – A045	マシン・スタック
A04A – A07F	インデックス・レジスタ スタック

　23．FN 文法エラー，FN 未定義　@
　24．STRING の使用法エラー　@
　25．ストリング・バッファ・オーバーフロー，又はストリングスの重なり　@
　26．入出力要求が出されたがＩ/Ｏカードが設置されていない@
　27．VAL 関数エラー　ストリングが数値以外の文字で始まった。　@
　28．LOG エラー　負の数の LOG　@

プログラムのSAVEとLOAD

SAVE

　SAVE コマンドは現在の BASIC プログラムをカセットか紙テープに出力します。SAVE の使い方は，MIKBUG のＰコマンドに似ています。パンチオン／オフの指令は自動的に装置に出されます。SWTPC AC-30 カセット・インターフェイスを使うときは，MANUAL 又は AUTOMATIC モータ・コントロールが選べます。レコーダを RECORD にし，SAVE をキーインすると，プログラムはカセットテープにダンプされます。SAVE コマンドは BASIC バッファ全体をダンプします。SAVE　10－20　などと，行番号を入れることはできません。SAVE オペレーションを行なってもメモリ内の BASIC プログラムはそのまま残っています。

LOAD

　LOAD コマンドは，カセット又は紙テープに記録されたプログラムをロードするために使われます。LOAD は，MIKBUG のＬコマンドに以ています。リーダオン／オフ指令は自動的に送られます。LOAD とタイプすると，プログラムはテープから BASIC バッファへ読み込まれます。LOAD が始まる時に，バッファは自動的にクリヤされます。

　LOAD コマンドでは，１文字のファイル名を使うことができます。例えば，LOAD　B　とすると，SAVE　B　でセーブされたプレグラムだけがロードされます。ファイル名を省略すると，テープ上のプログラムが名前に関係なくロードされます。LOAD（ファイル名）コマンドを使うとき，テープリーダは，READER ON モードに固定されなければなりません。

　NOTE：SAVE 及び LOAD コマンドは，パンチ／リード襲置がリーダ／パンチ／オン／オフを自動的にデコードすると仮定しています　もし，あなたの装置が自動でなかったら，リーダ又はパンチは LOAD, SAVE の後の CR を押す前に手動でロックしておかなければなりません。

BASICのローディング

　BASIC インタプリタ紙テープ及びカセットテープのどちらでも使えるように作られています。BASIC をロードする前にメモリ８Ｋ（0000～1FFF）あるかどうか確認して下さい。BASIC をＳＷＴＰＣ AC－30 カセット・インタフェイス又は紙テープからロードするのは，他のプログラムをロードするのと同じように行ないます　提供されるテープは BASIC インタプリタをロードし，同時にプログラム・カウンタ・アドレスＡ048とＡ049を BASIC の開始番地0100にセットします。スタートさせるには，Ｇ(GOTO User Program）を押します

　何らかの事情で、RESET ボタンを押したとき，前に入れたプログラムが残っているようにするためにはＡ048とＡ049を0103にしてから，Ｇを押します。プログラム・カウンタを0100にして，BASICインタプリタにもどると，前に入れたプログラムは消去されます。

謝　辞

1.　スロットマシンゲーム(8080)

　機材の提供をしていただいた〝ムーン・ベース〟に感謝いたします。

2.　6800　4K　BASIC

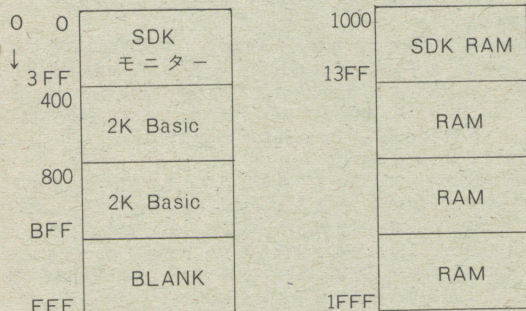
　プログラムの発表を許可されたROBERT H. UTTERWYK氏，機材の提供等に全面的な協力をしていただいた，サウスウエスト社およびサウスウエスト・ジャパン社，およびレコード製作上御指導いただいた石木氏に感謝いたします。

3.　8080　2K　BASIC

　プログラムの発表を許可された石田東大助教授，実際の開発にあたられた小野氏(東大)，レコード製作上御協力いただいた，パネトロン，Bit-Inn の皆様に感謝いたします。

(I/O 編集部)

8080 2K BASIC

メモリーマップ

I/O: TTY (ASR-33)

2K BASIC 文法

コマンド

NEW
LIST
RUN
SIZE

ステイトメント

LET
PRINT
INPUT
GOTO
GOSUB
RETURN
IF
FOR
NEXT
END

ファンクション

RND(<exp>)	=
ABS(<exp>)	#
	>
A~Z	>=
	<
@(<exp>)	<=

リテラル(Print用)

"string"
'string'
<exp>
←
#<exp>

8080 2K BASICについてのコメント　福島 慎

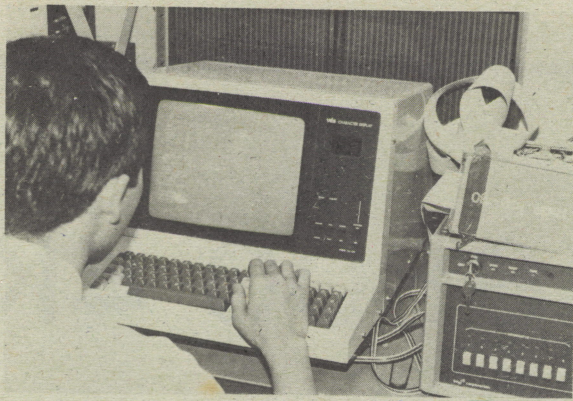
DDJ, Interface Age……と，どう，う具合なのか，Tiny Basicがあきもせずにでてきます．ところが最大の関心事であるI/OがどれもACIAをつかったTTY向きのインターフェイスなのです．

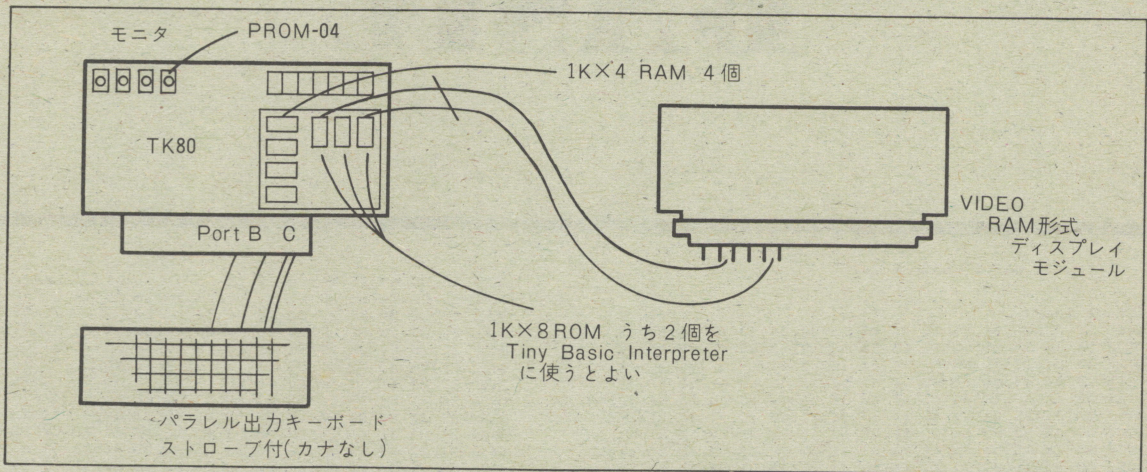
無論シリアルならボーレイトを変更して 300～4800 bpsのCRTターミナルでもよいでしょうが，もともとデータバスはパラレルで最終的にプリントしたりディスプレイしたり，あるいはキーボードから得られる信号もパラレルなのに何でシリ・パラ変換を２度もやるんだとご不満をお感じの方も多いと思います．(線の本数が少ない方がよいというプロの方には申し訳ありません) そこでパラレルボードとVideo-RAM形式のディスプレイ用にルーチンを差し換えた例をあげておきます．

```
TTY01: JNZ   TTY00
       POP   PSW
       RET
TTY00  CALL  WAIT
       NOP
       POP   PSW
       PUSH  PSW
       CALL  OUT33 (0344H)
       POP   PSW        PROM04サブルーチン
       CPI   0DH
       RNZ
       MVI   0AH
       CALL  PUT
       MVI   A, 0DH
       RET
```

```
BREAK: IN    02H……つまり8255 port C
       ANI   01H……つまりbit0 ——ストローブ
       RZ
       CALL  TTYI3
       ANI   7FH
       CPI   0FH
       JNZ   TTYI2
       LDA   OMASK
       CMA
       STA   OMASK
       JMP   BREAK
TTYI2: CPI   02H
       RNZ
       JMP   START
TTYI3: PUSH  B
       PUSH  D
       PUSH  H
       IN    01H
       MVI   E, 0H
       JMP   IN13 (0361H) PROM04内サブルーチン)
WAIT   PUSH  D
       CALL  D3 (02EFH) TK80モニタ内サブルーチン
       POP   D
       RET
```

```
ORG   IENT
MVI   A, 93H
OUT   03H
NOP
NOP
NOP
NOP
```

おっと忘れちゃいけない最初の方にも若干変更があります．(右上参照)

MDS ICE80(インサーキットエミュレーター）を用いて、SDK80キット上のTiny BASICのインプリンターのデバッグを行っているところ

＊申しおくれましたが使用するハードウェアは図の通りです．
＊インタープリンタをＲＡＭに入れる場合は0400～0BFFにＲＡＭを置けるように改造するとよいでしょう．

＊参考までにいっておくとＤ３はTK-80モニタ内のサブルーチンでDEレジスタをつかった27m/secのWAITルーチンでOUT33，IN13，はPROM-04内のサブルーチンです．（本誌にその記事がのっています．)〔KB-02 TVD-02 PROM-04いずれもアドテック製〕

TK80にKB-02, TVD-42を使用しましたがこれは自作キーボード キャラクタディスプレイにすぐ置きかえられるというだけのことに他ならないのです．

ここまで読んで賢明な皆さんは『なあ～んだたったそれだけのことか』と思われたかもしれませんが本当にその通りです．皆さんがTiny Basicのシステムを作る際の参考となれば幸いです．

最後になりましたが，今回Tiny Basicを手に入れようとほっつきまわった際お世話になったBit-Inn，アドテック，ASR，そしてI/Oの編集部の方に感謝しております．

■レコードを使う方へ

このプログラムは110 ボーで入っています．プログラムは400₁₆から始まっていますのでSDK80を持っている方はそのまま使えます．その他の方で例えばTTYに打出して見たい人はCMTインターフェイス(Bit-Innで売っているIC-0006，￥6,500)とTTYインターフェイスをつなげば紙テープで再生できます．その他，TK-80などをお持の方もRAMを増設して，利用してください．

編集後記

■BINARY No.2はいかがでしたでしょうか 今回は神谷君のスロットマシンプログラムでしたが，今後もゲームだけでなく様々なソフトフェアを提供していきます．機械語レベルのものばかりでなく，BASICのプログラムも加えていきたいと考えています．読者の方も，発表してみたいプログラムがあれば，ぜひI/O編集部まで御連絡下さい．

■去る７月20日，新宿で首都圏の大学４校と他に１グループが集まってマイコン愛好者会合を開きましたが，なかなか活発に議論がかわされ，８月10日に，バス規格に関する技術会合（この原稿を書いている時点ではまだですが）を開くことを約束して終わりました．

他にこのような連合団体を作っている人，作りたいと思っているグループがありましたら，ぜひ御連絡下さい．

SWTPC 6800 4K

BASIC

```
SWTPC 4K BASIC VERSION 2.0

D 0100 1042
0100 7E 08 0E 7E 08 15 54 4F 1E 99 99 53 54 45 50 1E
0110 88 88 52 55 4E 1E 08 76 4C 49 53 54 1E 08 9D 4E
0120 45 57 1E 08 0E 50 41 54 1E 08 EE 4C 4F 41 1E 09
0130 28 53 41 56 1E 09 00 47 4F 53 55 42 1E 0A 6A 47
0140 4F 54 4F 1E 0A 8C 4F 4E 1E 0A 4B 54 48 45 4E 1E
0150 0A 39 50 52 49 4E 54 1E 0C 82 4C 45 54 1E 0D D5
0160 49 4E 50 55 54 1E 0A D7 49 46 1E 0F E0 52 45 41
0170 44 1E 0C 2B 44 41 54 41 1E 0D F9 52 45 53 54 4F
0180 52 45 1E 0C 79 45 4E 44 1E 08 15 52 45 54 55 52
0190 4E 1E 0A B2 44 49 4D 1E 0E 21 46 4F 52 1E 0E A2
01A0 4E 45 58 54 1E 0F 53 52 45 4D 1E 0D F9 41 50 50
01B0 1E 09 2B 53 54 4F 50 1E 0A C6 1E 0D D5 52 4E 44
01C0 28 1E 10 BE 54 41 42 28 1E 0C C7 49 4E 54 28 1E
01D0 10 7C 43 48 52 24 28 1E 0C BA 43 48 52 28 1E 0C
01E0 BA 41 42 53 28 1E 10 B5 53 47 4E 28 1E 10 9A 55
01F0 53 45 52 28 1E 10 6E 13 14 0D 0A 15 52 45 41 44
0200 59 1E 20 44 45 4C 1E 10 16 1E 52 45 2D 45 4E 54
0210 45 52 1E 45 52 52 4F 52 23 20 1E 20 49 4E 20 4C
0220 49 4E 45 20 1E CE 02 02 8D 6A 20 07 86 3F 8D 48
0230 BD 03 59 CE 00 B0 8D 45 81 18 27 E9 81 0D 27 25
0240 81 0F 26 0C 86 5F 8D 30 8C 00 B0 27 E9 09 20 E6
0250 8C 00 F8 26 05 C6 21 7E 08 47 A7 00 08 20 D7 86
0260 23 8D 15 20 CE 86 1E A7 00 DF AE 96 85 26 02 8D
0270 31 39 7E E0 BF 7E E0 C8 8D 09 7E E1 D1 BD E1 AC
0280 36 20 08 36 B6 80 04 2B 09 8D F2 81 03 26 03 7E
0290 08 15 32 39 8D 05 20 0A 8D DE 08 A6 00 81 1E 26
02A0 F7 39 8D 10 CE 02 AC 8D F2 8D 1E 39 0D 0A 15 FF
02B0 FF FF FF 1E DF 38 DE 3A 09 09 DF 3A 36 96 38 A7
02C0 00 96 39 A7 01 32 DE 38 39 DE 3A EE 00 7C 00 3B
02D0 7C 00 3B 39 8D DE BD 04 4E 9F 28 35 DE 3A EE 00
02E0 20 11 8D D0 BD 04 4E 9F 28 35 09 09 09 09 09 09
02F0 09 EE 00 C6 05 32 A7 00 08 5A 26 F9 32 32 A7 00
0300 9E 28 8D C5 39 8D AD 9F 28 35 20 0A 8D A6 BD 04
0310 4E 9F 28 AE 00 34 C6 05 DE 5D 32 A7 00 08 5A 26
0320 F9 32 A7 01 6F 00 9E 28 BD 04 47 8D 9C 39 BD 02
0330 75 A6 00 08 BD 02 B4 CE 01 12 DF 48 97 49 DE 48
0340 09 09 A6 00 81 1E 26 F9 08 08 08 BD 02 9B 8D 09
0350 7F 00 40 BD 02 C9 7E 02 94 36 86 20 BD 0D 0B 32
0360 39 BD 07 60 BD 04 33 25 26 16 A6 01 BD 04 33 25
0370 20 BD 07 A5 BD 02 B4 DE 34 BD 04 BA DF 34 BD 02
0380 C9 EE 00 AD 00 BD 04 47 DE 34 31 31 31 31 39 0B
0390 39 BD 04 3B 24 07 81 28 27 03 09 86 20 08 08 BD
03A0 02 B4 DE 2A 9C 46 0D 27 09 E1 00 26 0C A1 01 26
03B0 08 0C DF 63 BD 02 C9 0A 39 36 A6 01 81 28 32 27
03C0 05 BD 04 5D 20 DE EE 04 20 DA BD 02 83 8D 92 28
03D0 01 39 36 25 31 DF 34 DE 63 81 28 27 15 08 08 DF
03E0 63 DE 5D 96 63 A7 00 96 64 A7 01 8D 5A 32 DE 34
03F0 0C 39 BD 0D FF 24 04 8D 74 20 01 5F 37 8D 6E 17
0400 33 BD 0E 75 20 D9 DF 34 DE 63 E7 00 A7 01 81 28
0410 26 19 C6 0A E7 02 E7 03 BD 0D FF 25 03 5F E7 03
0420 A6 02 BD 0E 5D 6D 03 27 D2 20 CC 8D 30 DF 46 DE
0430 63 20 AA 81 41 2B 0C 81 5A 2F 0A 81 30 2B 04 81
0440 39 2F 02 0D 39 0C 39 DE 5D 8D 13 DF 5D 39 DE 5D
0450 8D 03 20 F7 09 09 09 09 09 09 09 09 39 08 08 08
0460 08 08 08 08 08 9C 44 2B 03 7E 0A 0B 39 BD 02 B4
0470 8D DC A6 06 81 04 2F 05 C6 01 7E 08 47 4F 5F 6D
0480 06 23 19 58 49 D7 52 97 51 58 49 58 49 DB 52 99
0490 51 EB 00 89 00 BD 05 5C 6A 06 20 E3 4D 26 D9 BD
04A0 02 C9 39 BD 07 60 BD 03 CA 25 04 BD 03 0C 39 BD
04B0 0B 3B 25 01 39 81 28 26 11 08 8D 33 BD 07 60 81
04C0 29 26 02 08 39 C6 13 7E 08 47 C6 06 20 F9 8D D3
04D0 BD 07 60 81 2A 26 0B 08 BD 07 60 8D C6 BD 06 E6
04E0 20 EE 81 2F 26 08 08 8D BA BD 06 6E 20 E2 39 BD
04F0 07 60 81 2D 26 07 08 8D D5 8D 2D 20 07 81 2B 26
0500 01 08 8D CA BD 07 60 81 2B 26 08 08 8D C0 BD 06
0510 07 20 F1 81 2D 26 08 08 8D B4 BD 06 04 20 E5 39
0520 BD 02 B4 BD 04 5E 20 0C BD 02 B4 DE 5D 09 09 20
0530 06 BD 02 B4 BD 04 60 36 37 C6 06 86 99 A0 00 A7
0540 00 09 5A 26 F6 BD 04 5F 0D C6 06 86 00 A9 00 19
0550 A7 00 09 5A 26 F5 BD 02 C9 33 32 39 36 8D 08 A6
0560 00 84 0F A7 00 32 39 36 37 DF 57 C6 04 DE 57 BD
0570 04 60 68 00 86 05 09 69 00 4A 26 FA 5A 26 EE 33
0580 32 39 36 37 DF 57 A6 00 84 F0 97 53 C6 04 DE 57
0590 64 00 86 05 08 66 00 4A 26 FA 5A 26 F1 DE 57 A6
05A0 00 9B 53 A7 00 33 32 39 BD 02 B4 BD 04 4E 7F 00
05B0 56 BD 0D BC 24 19 A6 00 2A 0B 97 56 BD 05 31 86
05C0 0F A4 00 A7 00 A6 00 26 08 8D 91 6A 06 20 F6 6F
05D0 06 6F 05 A6 06 81 65 2D 15 37 C6 05 86 99 A7 00
05E0 08 5A 26 FA 86 64 A7 01 BD 04 57 33 20 D1 81 9D
05F0 2E 03 BD 0B F0 7D 00 56 27 03 BD 05 31 BD 04 47
0600 BD 02 C9 39 BD 05 28 BD 02 B4 BD 04 4E BD 0D BC
0610 24 28 BD 05 82 6C 06 BD 04 4E BD 0D BC 24 11 BD
0620 05 82 6C 06 A6 06 A1 0D 27 08 2D F3 8D 10 20 EF
0630 8D 0C 8D 23 BD 04 47 BD 05 A8 BD 02 C9 39 DE 5D
0640 37 36 C6 07 A6 00 36 A6 07 A7 00 32 A7 07 08 5A
0650 26 F2 DE 5D 32 33 39 36 37 DE 5D C6 06 0C A6 05
0660 A9 0C 19 A7 05 09 5A 26 F5 DE 5D 33 32 39 BD 02
0670 B4 BD 0B F0 BD 04 4E BD 0D BC 25 05 C6 08 7E 08
0680 47 60 06 8D 38 BD 05 82 BD 05 20 86 0B 5F 5C 8D
0690 C6 6D 00 2A F9 5A BD 05 20 8D BC BD 05 20 BD 04
06A0 5E BD 04 5E BD 05 67 EB 05 E7 05 DE 5D BD 05 67
06B0 4A 26 DA BD 04 47 8D 86 7C 00 50 20 59 BD 05 82
06C0 4F 6D 00 2A 05 BD 05 31 86 80 BD 04 4E 6D 00 2A
06D0 05 BD 05 31 8B 80 97 4A A6 06 AB 0D 28 05 86 78
06E0 24 01 40 97 50 39 BD 02 B4 BD 04 4E 8D CF DF 59
06F0 BD 04 47 BD 06 3E BD 0B F0 86 09 DE 59 E6 00 BD
0700 05 5C 5D 27 06 BD 06 57 5A 20 F7 DE 5D BD 04 5E
0710 BD 05 82 4A 26 E5 BD 04 4E BD 06 3E 96 50 A7 06
0720 BD 04 47 BD 05 A8 7D 00 4A 2A 03 BD 05 28 BD 02
0730 C9 39 96 32 D6 33 D0 31 92 30 25 1E DE 2E 96 30
0740 D6 31 E0 01 A2 00 25 14 26 03 5D 27 10 08 08 A6
0750 00 81 1E 26 F9 08 9C 2A 26 E4 DE 2A 0D DF 2C 39
0760 A6 00 81 20 26 03 08 20 F7 39 BD 0B 3B 24 05 C6
0770 07 7E 08 47 DF AC BD 04 4E E6 06 23 F2 C1 04 2E
0780 EE 86 05 10 4D 27 06 BD 05 82 4A 20 F7 A6 01 97
0790 30 A6 02 97 31 DE AC 0C 39 DE 34 A6 00 08 81 1E
07A0 26 F9 DF 36 39 DF AC 9F 28 CE 01 BC 20 12 DF AC
07B0 9F 28 CE 01 05 20 09 8D A7 DF AC 9F 28 CE 01 11
07C0 9E AC 34 08 32 81 20 27 FB E6 00 C1 1E 27 18 11
07D0 27 F1 08 8C 01 BA 27 17 8C 01 F4 27 18 E6 00 C1
07E0 1E 26 EF 08 08 20 D9 08 9F AC 9F 34 9E 28 39 9E
07F0 28 CE 01 BB 39 7E 0B 2B CE 12 00 DF 2E DF 2A DF
0800 46 4F 97 32 97 33 97 40 CE 10 7B DF 67 39 8E A0
```

```
0810 45 8D E5 20 61 8E A0 45 CE A0 7F DF 3A 7F 00 85
0820 CE 01 F7 BD 02 94 7F 00 40 BD 02 5F CE 00 B0 BD
0830 07 60 BD 04 3B 25 05 BD 09 59 20 EA 81 1E 27 E6
0840 BD 07 B7 EE 00 6E 00 8E A0 45 BD 02 A2 CE 02 13
0850 BD 02 9B D7 20 CE 00 20 BD 02 72 CE 02 1B BD 02
0860 9B DE 5B 96 34 26 07 CE 00 30 6F 00 6F 01 BD 02
0870 75 BD 02 A2 20 9F DE 2E DF 3C DF 36 CE 00 75 DF
0880 65 CE 11 80 DF 5F DE 2A DF 46 4F 09 08 A7 00 9C
0890 44 27 07 A6 00 27 F5 09 DF 44 7E 0A 10 CE 02 07
08A0 BD 02 94 DE 34 BD 07 60 81 1E 27 2B BD 07 6A DF
08B0 34 BD 07 3C DF 24 DE 34 BD 07 60 81 1E 27 04 08
08C0 BD 07 6A 4F C6 01 DB 31 19 D7 31 99 30 19 97 30
08D0 BD 07 3C DE 24 20 06 DE 2A DF 2C DE 2E 9C 2C 27
08E0 0A 9C 2A 27 06 BD 03 2E 08 20 F2 7E 0D F9 BD 07
08F0 99 CE 0A 10 FF A0 46 8E A0 40 BF A0 08 7E E0 E3
0900 DE 2E 86 12 BD 02 78 8D 3A 8D 38 8D 41 9C 2A 27
0910 0D 86 02 BD 02 78 BD 03 2E 08 8D 29 20 EF 86 03
0920 BD 02 78 8D 29 7E 08 15 BD 07 F8 86 11 97 85 BD
0930 02 78 BD 02 7D 81 02 26 F9 BD 02 33 CE 00 B0 8D
0940 18 20 EF 8D 00 5F 4F 4A 26 FD 5A 26 F9 39 C6 32
0950 86 FF BD 02 78 5A 26 F8 39 BD 07 6A BD 07 32 24
0960 13 DE AC BD 07 60 81 1E 27 21 DE 2C 9C 2A 27 1C
0970 8D 48 20 17 DE AC BD 07 60 81 1E 26 0A DE 2A 9C
0980 2E 27 08 8D 0F 20 04 8D 0B 8D 2F 39 8D 2C DE 30
0990 DF 32 20 F7 9F 28 DE 2C 9E 2A 08 08 34 34 A6 00
09A0 34 08 81 1E 26 F8 9F 2A 9F 46 35 DE 2C 9C 2A 27
09B0 06 32 A7 00 08 20 F6 9E 28 39 DE AC BD 07 B7 9F
09C0 28 DF 48 DE 2A DF 20 D6 AF D0 AD CB 04 DB 2B 86
09D0 00 99 2A 91 44 22 34 D7 2B 97 2A DE 2A DF 46 35
09E0 DE 20 9C 2C 27 06 09 A6 00 36 20 F6 DE 2C 96 30
09F0 A7 00 08 96 31 A7 00 08 96 49 A7 00 08 9E AC 34
0A00 32 A7 00 08 81 1E 26 F8 9E 28 39 C6 14 7E 08 47
0A10 CE 01 12 DF 48 CE 11 02 DF 5D DE 36 DF 5B 9C 2A
0A20 26 03 7E 08 15 7D 00 36 27 F8 00 08 A6 00 08 DF
0A30 34 97 49 DE 48 EE 00 6E 00 DE 34 BD 07 60 BD 04
0A40 3B 24 49 7E 0F F5 C6 20 7E 08 47 DE 34 BD 04 EF
0A50 BD 04 6D 5D 27 F0 5A D7 55 BD 07 B7 EE 00 8C 0A
0A60 8C 27 2C 8C 0A 6A 27 05 20 DC 7F 00 55 DE 36 BD
0A70 07 99 DE 65 8C 00 85 26 05 C6 09 7E 08 47 96 36
0A80 A7 00 08 96 37 A7 00 08 DF 65 20 03 7F 00 55 DE
0A90 34 BD 07 6A 7A 00 55 2B 0A BD 07 60 81 2C 26 A6
0AA0 08 20 EE BD 07 32 24 05 C6 07 7E 08 47 DF 36 7E
0AB0 0A 10 DE 65 8C 00 75 26 05 C6 10 7E 08 47 09 09
0AC0 DF 65 EE 00 20 E7 CE 01 B3 BD 02 9B BD 03 59 DE
0AD0 36 BD 02 75 7E 08 15 96 34 26 05 C6 02 7E 08 47
0AE0 BD 02 2C CE 00 B0 DF AC DE 34 BD 03 CA 25 31 DF
0AF0 34 DE AC 8D 3B 24 12 81 1E 27 06 CE 02 0A BD 02
0B00 94 BD 02 2C CE 00 B0 20 EA BD 02 E2 BD 07 60 81
0B10 2C 26 01 08 DF AC DE 34 BD 07 60 08 81 2C 27 CA
0B20 09 7F 00 40 81 1E 26 03 7E 0D F9 C6 03 7E 08 47
0B30 BD 07 60 97 4A 08 81 2D 27 04 09 7F 00 4A 4F 97
0B40 4C 97 4D 97 4E BD 0B F0 D6 5D D7 26 D6 5E D7 27
0B50 C6 05 BD 07 60 BD 04 3B 24 0B 81 2E 26 05 97 4D
0B60 08 20 EF 0D 39 09 08 A6 00 BD 04 3B 25 42 80 30
0B70 26 0F 7D 00 4C 26 0C 7D 00 4D 27 EA 7A 00 4B 20
0B80 E5 97 4C BD 02 B4 DE 26 5D 27 16 7D 00 4E 26 08
0B90 AB 00 A7 00 08 5A 20 06 48 48 48 48 A7 00 73 00
0BA0 4E DF 26 7D 00 4D 26 03 7C 00 4B BD 02 C9 20 B6
0BB0 7D 00 4D 26 08 81 2E 26 04 97 4D 20 A9 5F 81 45
0BC0 26 14 08 A6 00 81 2D 27 09 81 2B 26 01 08 8D 36
0BD0 20 04 08 8D 31 50 96 4B 1B BD 02 B4 DE 5D A7 06
0BE0 BD 04 47 BD 02 C9 7D 00 4A 27 03 BD 05 28 0C 39
0BF0 BD 02 B4 37 C6 07 DE 5D 6F 00 08 5A 26 FA 7F 00
0C00 4B 33 BD 02 C9 39 A6 00 BD 04 3B 24 05 C6 02 7E
0C10 08 47 80 30 16 08 A6 00 BD 04 3B 25 0D 58 17 58
0C20 58 1B 16 A6 00 80 30 1B 16 08 39 DE 34 BD 03 CA
0C30 25 3F DF 34 DE 3C 9C 2E 27 1C BD 0B 30 24 22 09
0C40 A6 00 81 1E 27 05 C6 19 7E 08 47 9C 2A 27 F7 A6
0C50 00 08 81 1E 26 F5 A6 02 81 79 26 EF 08 08 08 20
0C60 D9 BD 02 E2 08 DF 3C DE 34 BD 07 60 08 81 2C 27
0C70 BC 09 81 1E 26 D0 7E 0D F9 DE 2E DF 3C DE 34 7E
0C80 0D F9 DE 34 BD 07 60 81 22 26 14 08 A6 00 08 81
0C90 22 27 4A 81 1E 26 04 C6 04 20 1C 8D 6E 20 ED 81
0CA0 1E 26 35 09 A6 00 08 81 3B 27 06 BD 02 A2 7F 00
0CB0 40 08 DF 36 7E 0A 10 7E 08 47 DE 34 BD 04 6D 17
0CC0 8D 49 8E A0 45 20 16 DE 34 BD 04 6D D0 40 23 F2
0CD0 BD 03 59 5A 26 FA 20 EA BD 04 EF 8D 4B BD 07 60
0CE0 81 2C 26 0E 08 96 40 16 C4 F0 10 27 0A BD 03 59
0CF0 20 F3 81 3B 26 04 08 7E 0C 84 81 1E 27 A1 C6 06
0D00 20 B5 86 2E 20 05 4F 84 0F 8B 30 36 BD 02 78 32
0D10 37 D6 40 5C C1 30 23 0C C1 3F 27 04 81 20 26 04
0D20 BD 02 A2 5F D7 40 33 39 BD 02 B4 BD 04 4E 4F 97
0D30 4D 6D 00 26 08 8D CF BD 02 C9 7E 03 59 6D 00 2A
0D40 07 86 2D 8D C6 BD 05 31 A6 06 81 09 2E 31 81 FF
0D50 2D 2D 4D 2E 10 8D AF 8D A9 97 4D 6D 06 2A 1C 8D
0D60 A5 6C 06 20 F6 7D 00 4D 26 11 A6 00 8D 99 BD 05
0D70 5C 6A 06 26 F5 8D 45 24 BE 8D 87 8D 30 20 B8 A6
0D80 00 8D 84 BD 05 5C BD 0D 02 8D 22 86 45 BD 0D 0B
0D90 E6 06 5A 2A 06 86 2D BD 0D 0B 50 4F 4C C0 0A 2A
0DA0 FB 4A CB 0A BD 0D 07 17 BD 0D 07 20 8A 8D 0D 24
0DB0 0A A6 00 BD 0D 07 BD 05 5C 20 F2 39 BD 02 B4 DE
0DC0 5D 37 C6 06 6D 00 26 07 08 5A 26 F8 0C 20 01 0D
0DD0 BD 02 C9 33 39 DE 34 BD 03 CA 24 05 C6 12 7E 08
0DE0 47 BD 07 60 08 81 3D 27 04 C6 06 20 F1 BD 04 EF
0DF0 81 1E 26 F5 BD 02 E2 20 00 BD 07 99 7E 0A 10 BD
0E00 02 B4 DE 34 BD 04 EF 08 81 2C 27 0B 81 29 26 1A
0E10 DF 34 BD 02 C9 0C 39 BD 04 BA DF 34 BD 02 C9 0D
0E20 39 DE 34 BD 03 61 29 02 25 05 C6 05 7E 08 47 81
0E30 28 26 F7 DF 34 DE 63 E7 00 A7 01 8D C2 24 05 BD
0E40 04 6D 20 01 5F E7 03 BD 04 6D 17 A7 02 8D 0E DE
0E50 34 BD 07 60 08 81 2C 27 CA 09 7E 0D F9 BD 02 B4
0E60 E6 03 8D 11 BD 04 5F DF 46 BD 02 C9 96 46 A7 04
0E70 96 47 A7 05 39 4D 26 05 C6 15 7E 08 47 A1 02 22
0E80 F7 E1 03 22 F3 36 A6 02 97 51 5D 27 0D 5A 27 0A
0E90 96 51 BD 04 5F 4A 26 FA 20 F3 32 BD 04 5F 4A 26
0EA0 FA 39 DE 34 BD 03 CA 24 03 7E 0D DC DF 34 DE 5D
0EB0 BD 04 55 A6 00 E6 01 CE 11 80 9C 5F 27 0F A1 00
0EC0 26 04 E1 01 27 07 BD 04 5D 20 EF DE 5F 8C 12 00
0ED0 26 05 C6 10 7E 08 47 A7 00 08 E7 00 08 DF 5F DE
0EE0 34 BD 07 60 08 81 3D 27 03 7E 0D E9 BD 04 EF BD
0EF0 02 E2 BD 07 AE 8C 01 09 26 EF DE 34 BD 04 EF DF
0F00 34 DE 5F BD 02 D4 BD 04 5F DF 5F DE 34 BD 07 AE
0F10 8C 01 10 26 09 DE 34 BD 04 EF DF 34 20 0E DE 5D
0F20 BD 0B F0 86 01 A7 00 A7 06 BD 04 47 DE 5F BD 02
0F30 D4 BD 04 5F DF 5F DE 34 BD 07 60 81 1E 26 AA 08
0F40 DF 36 DE 5F D6 36 E7 00 08 D6 37 E7 00 08 DF 5F
0F50 7E 0A 10 DE 34 BD 03 CA 24 03 7E 0D DC BD 07 60
0F60 81 1E 26 D9 08 DF 36 BD 04 4E A6 00 E6 01 DE 5F
0F70 8C 11 80 27 66 DF 5F 8C 11 80 27 5B BD 04 54 BD
0F80 04 54 A1 00 26 EF E1 01 26 EB DF 61 EE 00 BD 03
0F90 05 DE 61 BD 04 5D 86 02 6D 00 2A 02 8B 03 97 4F
0FA0 BD 03 05 BD 06 07 CE 00 6F BD 02 D4 BD 04 47 DE
0FB0 61 08 08 BD 03 05 BD 10 3A 24 07 DE 61 DF 5F 7E
0FC0 0A 10 CE 00 6F BD 03 05 DE 61 EE 00 BD 02 D4 DE
0FD0 61 EE 0E DF 36 20 E8 C6 18 20 02 C6 17 7E 08 47
0FE0 DE 34 BD 04 EF 8D 17 97 4F BD 04 EF DF 34 8D 4A
0FF0 24 03 7E 0D F9 DE 34 BD 07 B7 EE 00 6E 00 BD 07
1000 60 08 81 3D 26 03 86 00 39 E6 00 81 3C 26 13 C1
1010 3D 26 04 08 86 02 39 C1 3E 26 04 08 86 03 39 86
1020 01 39 81 3E 27 05 C6 06 7E 08 47 C1 3D 26 04 08
1030 86 05 39 C1 3C 27 E4 86 04 39 96 4F 48 48 CE 10
1040 54 DF 20 9B 21 97 21 BD 06 04 BD 04 4E A6 00 DE
1050 20 4D 6E 00 27 16 20 12 2B 12 20 0E 2B 0E 20 F4
1060 26 0A 20 06 27 04 2B 02 2A 02 0D 39 0C 39 BD 04
1070 4E BD 02 B4 DE 67 AD 00 BD 02 C9 39 BD 04 4E C6
1080 09 E0 06 5D 2F 08 BD 05 82 6C 06 5A 20 F5 6F 05
1090 BD 04 47 BD 05 A8 BD 04 4E 39 BD 04 4E A6 00 27
10A0 13 97 4A BD 0B F0 86 01 A7 00 A7 06 7D 00 4A 2A
10B0 03 BD 05 31 39 BD 04 4E 6D 00 2A F8 20 F3 BD 04
10C0 4E BD 0D BC 24 05 BD 04 47 20 06 CE 00 69 BD 03
10D0 05 CE 10 F9 BD 03 05 BD 06 E6 BD 04 4E BD 05 5C
10E0 A6 01 36 A6 03 A7 01 32 A7 03 6F 06 BD 04 47 BD
10F0 05 A8 CE 00 69 BD 02 D4 39 08 37 25 41 69 00 B5
1100 B6
```

東大版8080 2K BASIC
①アセンブル リスト

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 1


                        PALO ALTO TINY BASIC

  1400          STACK    EQU   1400H
  1300          VTOP     EQU   1300H
  1337          LBUF     EQU   1337H
  13A7          MSTK     EQU   13A7H
  0400          IENT      EQU   0400H
  1000          VENT      EQU   1000H
                TST      MACRO STR,ADRS
                         CALL  TEST
                         DB    STR
                         DB    ADRS-$-1
                         ENDM
                CASE     MACRO STR,ADRS
                         DB    STR
                         DB    (ADRS+8000H) SHR 8
                         DB    ADRS AND OFFH
                         ENDM
  0400                   ORG   IENT
  0400 F3       START:   DI
  0401 310014            LXI   SP,STACK
  0404 C3BC04            JMP   ENTRY

  0407 E3       TEST:    XTHL                        TEST CHARACTER
  0408 CD2604            CALL   SKIP
  040B BE                CMP   M
  040C C36804            JMP   TEST1

  040F 3E0D     CRLF:    MVI   A,0DH
  0411 F5       PUT:     PUSH  PSW
  0412 3A0010            LDA   OMASK
  0415 B7                ORA   A                 ;CLEAR CARRY
  0416 C3640B            JMP   TTY01

  0419 CD5908   EEXPR:   CALL  EXPR
  041C E5                PUSH  H
  041D C31508            JMP   RST31


  0420 7C       COMP:    MOV   A,H
  0421 BA                CMP   D
  0422 C0                RNZ
  0423 7D                MOV   A,L
  0424 BB                CMP   E
  0425 C9                RET
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 2


  0426 1A       SKIP:    LDAX  D
  0427 FE20              CPI   20H
  0429 C0                RNZ
  042A 13                INX   D
  042B C32604            JMP   SKIP
                ;
  042E CDB309   ENDL:    CALL  TSTSC
  0431 C3CC09            JMP   LERMS
                ;
                ;
  0434 CD2604   TSTV:    CALL  SKIP
  0437 D640              SUI   40H
  0439 D8                RC
  043A C25804            JNZ   TSTV1
  043D 13                INX   D
  043E CD0D09            CALL  EXPR0
  0441 29                DAD   H
  0442 DAA104            JC    ERROR
  0445 D5                PUSH  D
  0446 EB                XCHG
  0447 CD5709            CALL  RSIZE
  044A CD2004            CALL  COMP
  044D DAFC09            JC    SYS01
  0450 210013            LXI   H,VTOP
  0453 CD7A09            CALL  DIFF
  0456 D1                POP   D
  0457 C9                RET
                ;
  0458 FE1B     TSTV1:   CPI   1BH               ;'A' 'Z'?
  045A 3F                CMC
  045B D8                RC                      ;NO
  045C 13                INX   D
  045D 210013            LXI   H,VTOP
  0460 07                RLC
  0461 85                ADD   L
  0462 6F                MOV   L,A
  0463 3E00              MVI   A,00H
  0465 8C                ADC   H
  0466 67                MOV   H,A
  0467 C9                RET
                ;
  0468 23       TEST1:   INX   H
  0469 CA7304            JZ    TSTEQ
  046C C5                PUSH  B
  046D 4E                MOV   C,M               ;FETCH SKIP NUMBER
  046E 0600              MVI   B,00H
  0470 09                DAD   B
  0471 C1                POP   B
  0472 1B                DCX   D
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 3

0473 13        TSTEQ:  INX    D
0474 23                INX    H
0475 E3                XTHL
0476 C9                RET
              ;
0477 210000    GINT:   LXI    H,0000H       ;GET INTEGER
047A 44                MOV    B,H
047B CD2604            CALL   SKIP
047E FE30      GETI1:  CPI    30H
0480 D8                RC                   ;RETURN INTEGER IN HL
0481 FE3A              CPI    3AH
0483 D0                RNC
0484 3EF0              MVI    A,0F0H
0486 A4                ANA    H             ;IF INTEGER>#0FFF ERROR
0487 C2A104            JNZ    ERROR
048A 04                INR    B
048B C5                PUSH   B
048C 44                MOV    B,H           ;HL_HL*10
048D 4D                MOV    C,L
048E 29                DAD    H
048F 29                DAD    H
0490 09                DAD    B
0491 29                DAD    H
0492 1A                LDAX   D             ;HL_HL+(DE)&#0F
0493 13                INX    D
0494 E60F              ANI    0FH
0496 85                ADD    L
0497 6F                MOV    L,A
0498 3E00              MVI    A,00H
049A 8C                ADC    H
049B 67                MOV    H,A
049C C1                POP    B
049D 1A                LDAX   D
049E F27E04            JP     GETI1
              ;
04A1 D5        ERROR:  PUSH   D
04A2 11A804    ERR1:   LXI    D,HOWMS
04A5 C3D009            JMP    LEMS1
              ;
04A8 484F573F  HOWMS:  DB     'HOW?'
04AC 0D                DB     0DH
04AD 4F4B      OKMES:  DB     'OK'
04AF 0D                DB     0DH
04B0 57484154  WHTMS:  DB     'WHAT?'
04B4 3F
04B5 0D                DB     0DH
04B6 534F5252  SRYMS:  DB     'SORRY'
04BA 59
04BB 0D                DB     0DH
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 4

04BC CD0F04    ENTRY:  CALL   CRLF
04BF 11AD04            LXI    D,OKMES
04C2 97                SUB    A
04C3 CD700A            CALL   MSG
04C6 21CD04            LXI    H,ST2+1
04C9 220110            SHLD   COBJ
04CC 210000    ST2:    LXI    H,0000H
04CF 220710            SHLD   FCNTR
04D2 220310            SHLD   RSTCK
              ;
04D5 3E3E      GETC:   MVI    A,3EH         ;'>' PROMPT
04D7 CD020A            CALL   GETL
04DA D5                PUSH   D             ;SAVE LINE END
04DB 113713            LXI    D,LBUF
04DE CD7704            CALL   GINT          ;HL=LABEL NUMBER
04E1 CD2604            CALL   SKIP
04E4 7C                MOV    A,H
04E5 B5                ORA    L
04E6 C1                POP    B             ;RECOVER LINE END
04E7 CABF05            JZ     KWCPR         ;IF LABEL NUMBER=0 COMMAND
04EA 1B        INSRT:  DCX    D
04EB 7C                MOV    A,H
04EC 12                STAX   D
04ED 1B                DCX    D
04EE 7D                MOV    A,L
04EF 12                STAX   D
04F0 C5                PUSH   B          ;SAVE LINE BOTTOM
04F1 D5                PUSH   D          ;SAVE LINE TOP
04F2 79                MOV    A,C
04F3 93                SUB    E
04F4 F5                PUSH   PSW               ;SAVE LINE LENGTH
04F5 CD460A            CALL   SRCH
04F8 D5                PUSH   D          ;SAVE CURRENT LINE
04F9 C20C05            JNZ    MOVE
04FC D5                PUSH   D          ;EXIST SAME LINE
04FD CD640A            CALL   SKIPL
0500 C1                POP    B
0501 2A1310            LHLD   OBTM
0504 CD050B            CALL   TRNSF         ;ERASE OLD LINE
0507 60                MOV    H,B
0508 69                MOV    L,C
0509 221310            SHLD   OBTM          ;UPDATE BOTTOM POINTER
050C C1        MOVE:   POP    B             ;RECOVER CURRENT LINE
050D 2A1310            LHLD   OBTM
0510 F1                POP    PSW           ;RECOVER LINE LENGTH
0511 E5                PUSH   H             ;SAVE CURRENT BOTTOM
0512 FE03              CPI    03H
0514 CA0004            JZ     START         ;LENGTH=3? RESTART
0517 85             ADD    L               ;COMPUTE MOVE BOTTOM
0518 6F                MOV    L,A
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 5

0519 3E00              MVI    A,00H
051B 8C                ADC    H
051C 67                MOV    H,A
051D 110013            LXI    D,VTOP
0520 CD2004            CALL   COMP
0523 D2FB09            JNC    SYSOF
0526 221310            SHLD   OBTM
0529 D1                POP    D             ;RECOVER CURRENT BOTTOM
052A CD100B            CALL   TR2           ;MOVE FOR INSERTION
052D D1                POP    D             ;RECOVER LINE TOP
052E E1                POP    H             ;RECOVER LINE BOTTOM
052F CD050B            CALL   TRNSF         ;INSERT LINE
0532 C3D504            JMP    GETC
              ;
              +CMDKW:  CASE   '''LIST''',LIST
0535 4C495354+         DB     'LIST'
0539 86       +        DB     (00631H+8000H) SHR 8
053A 31       +        DB     00631H AND 0FFH

              +        CASE   '''RUN''',RUN
053B 52554E   +        DB     'RUN'
053E 86       +        DB     (0Q601H+8000H) SHR 8
053F 01       +        DB     00601H AND 0FFH

              +        CASE   '''NEW''',NEW
0540 4E4557   +        DB     'NEW'
0543 85       +        DB     (005F2H+8000H) SHR 8
0544 F2       +        DB     005F2H AND 0FFH

              +STMKW:  CASE   '''NEXT''',NEXT
0545 4E455854+         DB     'NEXT'
0549 87       +        DB     (00731H+8000H) SHR 8
054A 31       +        DB     00731H AND 0FFH

              +        CASE   '''LET''',LET
054B 4C4554   +        DB     'LET'
054E 88       +        DB     (00807H+8000H) SHR 8
054F 07       +        DB     00807H AND 0FFH

              +        CASE   '''IF''',IFSTM
0550 4946     +        DB     'IF'
0552 87       +        DB     (0078CH+8000H) SHR 8
0553 8C       +        DB     0078CH AND 0FFH

              +        CASE   '''GOTO''',GOTO
0554 474F544F+         DB     'GOTO'
0558 86       +        DB     (00620H+8000H) SHR 8
0559 20       +        DB     00620H AND 0FFH

              +        CASE   '''GOSUB''',GOSUB
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 6

055A 474F5355+         DB     'GOSUB'
055E 42       +
055F 86       +        DB     (0068FH+8000H) SHR 8
0560 8F       +        DB     0068FH AND 0FFH

              +        CASE   '''RETURN''',RETRN
0561 52455455+         DB     'RETURN'
0565 524E     +
0567 86       +        DB     (006B1H+8000H) SHR 8
0568 B1       +        DB     006B1H AND 0FFH

              +        CASE   '''REM''',REM
0569 52454D   +        DB     'REM'
056C 87       +        DB     (00786H+8000H) SHR 8
056D 86       +        DB     00786H AND 0FFH

              +        CASE   '''FOR''',FOR
056E 464F52   +        DB     'FOR'
0571 86       +        DB     (006CCH+8000H) SHR 8
0572 CC       +        DB     006CCH AND 0FFH

              +        CASE   '''INPUT''',INPUT
0573 494E5055+         DB     'INPUT'
0577 54       +
0578 87       +        DB     (007A7H+8000H) SHR 8
0579 A7       +        DB     007A7H AND 0FFH

              +        CASE   '''PRINT''',PRINT
057A 5052494E+         DB     'PRINT'
057E 54       +
057F 86       +        DB     (00649H+8000H) SHR 8
0580 49       +        DB     00649H AND 0FFH

              +        CASE   '''STOP''',STOP
0581 53544F50+         DB     'STOP'
0585 85       +        DB     (005FBH+8000H) SHR 8
0586 FB       +        DB     005FBH AND 0FFH

0587 88                DB     (CMDER+8000H) SHR 8
0588 01                DB     CMDER AND 0FFH
              +FNKW:   CASE   '''RND''',RND
0589 524E44   +        DB     'RND'
058C 89       +        DB     (0091EH+8000H) SHR 8
058D 1E       +        DB     0091EH AND 0FFH

              +        CASE   '''ABS''',FNABS
058E 414253   +        DB     'ABS'
0591 89       +        DB     (0094BH+8000H) SHR 8
0592 4B       +        DB     0094BH AND 0FFH
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 7


              +         CASE  '''SIZE''',RSIZE
0593 53495A45+          DB    'SIZE'
0597 89       +         DB    (00957H+8000H) SHR 8
0598 57       +         DB    00957H AND OFFH

0599 88                 DB    (FACT2+8000H) SHR 8
059A FC                 DB    FACT2 AND OFFH
              +KWTO:    CASE  '''TO''',FORTO
059B 544F     +         DB    'TO'
059D 86       +         DB    (006DCH+8000H) SHR 8
059E DC       +         DB    006DCH AND OFFH

059F 89                 DB    (LERMS+8000H) SHR 8
05A0 CC                 DB    LERMS AND OFFH
              +KWSTP:   CASE  '''STEP''',FSTEP
05A1 53544550+          DB    'STEP'
05A5 86       +         DB    (006E8H+8000H) SHR 8
05A6 E8       +         DB    006E8H AND OFFH

05A7 86                 DB    (FSTP1+8000H) SHR 8
05A8 EE                 DB    FSTP1 AND OFFH
              +ROPKW:   CASE  '''>=''',GE
05A9 3E3D     +         DB    '>='
05AB 88       +         DB    (0081BH+8000H) SHR 8
05AC 1B       +         DB    0081BH AND OFFH

              +         CASE  '''#''',NE
05AD 23       +         DB    '#'
05AE 88       +         DB    (00821H+8000H) SHR 8
05AF 21       +         DB    00821H AND OFFH

              +         CASE  '''>''',GT
05B0 3E       +         DB    '>'
05B1 88       +         DB    (00827H+8000H) SHR 8
05B2 27       +         DB    00827H AND OFFH

              +         CASE  '''=''',EQ
05B3 3D       +         DB    '='
05B4 88       +         DB    (00836H+8000H) SHR 8
05B5 36       +         DB    00836H AND OFFH

              +         CASE  '''<=''',LE
05B6 3C3D     +         DB    '<='
05B8 88       +         DB    (0082EH+8000H) SHR 8
05B9 2E       +         DB    0082EH AND OFFH

              +         CASE  '''<''',LT
05BA 3C       +         DB    '<'
05BB 88       +         DB    (0083CH+8000H) SHR 8
05BC 3C       +         DB    0083CH AND OFFH
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 8


05BD 88                 DB    (NOROP+8000H) SHR 8
05BE 42                 DB    NOROP AND OFFH

05BF 213405   KWCPR:    LXI   H,CMDKW-1
05C2 CD2604   NXTKW:    CALL  SKIP
05C5 D5                 PUSH  D              ;SAVE LINE TOP
05C6 1A       KWC1:     LDAX  D
05C7 13                 INX   D
05C8 FE2E               CPI   2EH
05CA CAE305             JZ    KWSRT
05CD 23                 INX   H
05CE BE                 CMP   M
05CF CAC605             JZ    KWC1
05D2 3E7F               MVI   A,7FH
05D4 1B                 DCX   D
05D5 BE                 CMP   M
05D6 DAEA05             JC    EXEQT          ;WHOLE KEYWORD?
05D9 23       KWSK1:    INX   H
05DA BE                 CMP   M
05DB D2D905             JNC   KWSK1
05DE 23                 INX   H
05DF D1                 POP   D              ;RECOVER LINE TOP
05E0 C3C205             JMP   NXTKW

05E3 3E7F     KWSRT:    MVI   A,7FH
05E5 23       KWSK2:    INX   H
05E6 BE                 CMP   M
05E7 D2E505             JNC   KWSK2

05EA 7E       EXEQT:    MOV   A,M            ;FETCH EXEC ADDR HIGH
05EB 23                 INX   H
05EC 6E                 MOV   L,M            ;FETCH EXEC ADDR LOW
05ED E67F               ANI   7FH
05EF 67                 MOV   H,A
05F0 F1                 POP   PSW            ;DUMMY POP UP
05F1 E9                 PCHL                 ;GO TO EACH ROUTINE

05F2 CDC609   NEW:      CALL  TSCR2
05F5 211510             LXI   H,OTOP
05F8 221310             SHLD  OBTM
05FB CDC609   STOP:     CALL  TSCR2
05FE C30004             JMP   START

0601 CDC609   RUN:      CALL  TSCR2
0604 111510             LXI   D,OTOP

0607 210000   RUN1:     LXI   H,0000H
060A CD4E0A             CALL  SRCH1          ;SEARCH MINIMUM LABEL OR NEXT LABEL
060D DA0004             JC    START
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 9


0610 EB       RUN2:     XCHG
0611 220110             SHLD  COBJ
0614 EB                 XCHG
0615 13                 INX   D              ;SKIP LABEL
0616 13                 INX   D

0617 CD7E0B   NXTGO:    CALL  BREAK
061A 214405             LXI   H,STMKW-1
061D C3C205             JMP   NXTKW

0620 CD1904   GOTO:     CALL  EEXPR
0623 D5                 PUSH  D              ;SAVE LINE POINTER
0624 CDC609             CALL  TSCR2
0627 CD460A             CALL  SRCH
062A C2A204             JNZ   ERR1
062D F1                 POP   PSW            ;DUMMY POP
062E C31006             JMP   RUN2

0631 CD7704   LIST:     CALL  GINT
0634 CDC609             CALL  TSCR2
0637 CD460A             CALL  SRCH
063A DA0004   LISTL:    JC    START          ;IF EMPTY OR EOF RESTART
063D CDF00A             CALL  WLINE
0640 CD7E0B             CALL  BREAK
0643 CD4E0A             CALL  SRCH1
0646 C33A06             JMP   LISTL

0649 0E06     PRINT:    MVI   C,06H
              ;
              +         TST   ''';''',PRNT1
064B CD0704   +         CALL  TEST
064E 3B       +         DB    ';'
064F 06       +         DB    00656H-$-1

0650 CD0F04             CALL  CRLF
0653 C31706             JMP   NXTGO

              +PRNT1:   TST   0DH,PRNT2
0656 CD0704   +         CALL  TEST
0659 0D       +         DB    0000DH
065A 06       +         DB    00661H-$-1

065B CD0F04             CALL  CRLF
065E C30706             JMP   RUN1

              +PRNT2:   TST   '''#''',PRNT3
0661 CD0704   +         CALL  TEST
0664 23       +         DB    '#'
0665 07       +         DB    0066DH-$-1
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 10


0666 CD1904             CALL  EEXPR
0669 4D                 MOV   C,L
066A C37306             JMP   PRNT4
066D CD7E0A   PRNT3:    CALL  PR10
0670 C38406             JMP   PRNT6
              +PRNT4:   TST   ''',''',PRNT5
0673 CD0704   +         CALL  TEST
0676 2C       +         DB    ','
0677 06       +         DB    0067EH-$-1

0678 CDB309             CALL  TSTSC
067B C36106             JMP   PRNT2

067E CD0F04   PRNT5:    CALL  CRLF
0681 C32E04             JMP   ENDL

0684 CD1904   PRNT6:    CALL  EEXPR
0687 C5                 PUSH  B
0688 CDAE0A             CALL  WINT
068B C1                 POP   B
068C C37306             JMP   PRNT4

068F CD3B0B   GOSUB:    CALL  PSHV
0692 CD1904             CALL  EEXPR
0695 D5                 PUSH  D              ;SAVE OBJ POINTER
0696 CD460A             CALL  SRCH
0699 C2A204             JNZ   ERR1
069C 2A0110             LHLD  COBJ
069F E5                 PUSH  H
06A0 2A0310             LHLD  RSTCK
06A3 E5                 PUSH  H
06A4 210000             LXI   H,0000H
06A7 220710             SHLD  FCNTR
06AA 39                 DAD   SP
06AB 220310             SHLD  RSTCK
06AE C31006             JMP   RUN2

06B1 CDC609   RETRN:    CALL  TSCR2
06B4 2A0310             LHLD  RSTCK
06B7 7C                 MOV   A,H
06B8 B5                 ORA   L
06B9 CACC09             JZ    LERMS          ;IF RSTACKTOP=0 ERROR
06BC F9                 SPHL
06BD E1                 POP   H
06BE 220310             SHLD  RSTCK
06C1 E1                 POP   H
06C2 220110             SHLD  COBJ
06C5 D1                 POP   D
06C6 CD1F0B             CALL  POPV
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 11


  06C9 C32E04             JMP   ENDL

  06CC CD3B0B   FOR:      CALL  PSHV
  06CF CD9A09             CALL  LTSUB
  06D2 2B                 DCX   H
  06D3 220710             SHLD  FCNTR
  06D6 219A05             LXI   H,KWTO-1
  06D9 C3C205             JMP   NXTKW

  06DC CD1904   FORTO:    CALL  EEXPR
  06DF 220B10             SHLD  FTOV
  06E2 21A005             LXI   H,KWSTP-1
  06E5 C3C205             JMP   NXTKW

  06E8 CD1904   FSTEP:    CALL  EEXPR
  06EB C3F106             JMP   FORO

  06EE 210100   FSTP1:    LXI   H,0001H
  06F1 220910   FORO:     SHLD  FSTPV
  06F4 2A0110             LHLD  COBJ
  06F7 220D10             SHLD  FLABL
  06FA EB                 XCHG
  06FB 220F10             SHLD  FOBJ
  06FE 010A00             LXI   B,000AH
  0701 2A0710             LHLD  FCNTR
  0704 EB                 XCHG
  0705 60                 MOV   H,B
  0706 68                 MOV   L,B
  0707 39                 DAD   SP
  0708 3E                 DB    3EH                 (SKIP NEXT INSTRUCTION)

  0709 09       FOR3:     DAD   B
  070A 7E                 MOV   A,M
  070B 23                 INX   H
  070C B6                 ORA   M
  070D CA2A07             JZ    FOR10
  0710 7E                 MOV   A,M
  0711 2B                 DCX   H
  0712 BA                 CMP   D
  0713 C20907             JNZ   FOR3
  0716 7E                 MOV   A,M
  0717 BB                 CMP   E
  0718 C20907             JNZ   FOR3
  071B EB                 XCHG
  071C 210000             LXI   H,0000H
  071F 39                 DAD   SP
  0720 44                 MOV   B,H
  0721 4D                 MOV   C,L
  0722 210A00             LXI   H,000AH
  0725 19                 DAD   D
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 12


  0726 CD100B             CALL  TR2
  0729 F9                 SPHL
  072A 2A0F10   FOR10:    LHLD  FOBJ
  072D EB                 XCHG
  072E C32E04             JMP   ENDL
                ;
  0731 CD3404   NEXT:     CALL  TSTV
  0734 DACC09             JC    LERMS
  0737 220510             SHLD  NCNTR
  073A D5       NEXT1:    PUSH  D
  073B EB                 XCHG
  073C 2A0710             LHLD  FCNTR
  073F 7C                 MOV   A,H
  0740 B5                 ORA   L
  0741 CACD09             JZ    LEMSO
  0744 CD2004             CALL  COMP
  0747 CA5407             JZ    NEXT2               ;NEXT VAR.=FOR VAR.?
  074A D1                 POP   D
  074B CD1F0B             CALL  POPV
  074E 2A0510             LHLD  NCNTR
  0751 C33A07             JMP   NEXT1
  0754 5E       NEXT2:    MOV   E,M
  0755 23                 INX   H
  0756 56                 MOV   D,M                 ;DE_OLD FOR VALUE
  0757 2A0910             LHLD  FSTPV
  075A E5                 PUSH  H                   ;HL_STEP VALUE
  075B 19                 DAD   D
  075C EB                 XCHG                      ;DE_NEW FOR VALUE
  075D 2A0710             LHLD  FCNTR
  0760 73                 MOV   M,E
  0761 23                 INX   H
  0762 72                 MOV   M,D                 ;FOR VAR._NEW VALUE
  0763 2A0B10             LHLD  FTOV
  0766 F1                 POP   PSW
  0767 B7                 ORA   A
  0768 F26C07             JP    NEXT4
  076B EB                 XCHG
  076C CD9009   NEXT4:    CALL  CMINT
  076F D1                 POP   D
  0770 DA8007             JC    NEXT5
  0773 2A0D10             LHLD  FLABL
  0776 220110             SHLD  COBJ
  0779 2A0F10             LHLD  FOBJ
  077C EB                 XCHG
  077D C32E04             JMP   ENDL

  0780 CD1F0B   NEXT5:    CALL  POPV
  0783 C32E04             JMP   ENDL

  0786 210000   REM:      LXI   H,0000H
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 13


  0789 C39407             JMP   IFST2
                ;
  078C CD1904   IFSTM:    CALL  EEXPR
  078F 7C                 MOV   A,H
  0790 B5                 ORA   L
  0791 C21706             JNZ   NXTGO
  0794 CD660A   IFST2:    CALL  SKPL2
  0797 D21006             JNC   RUN2
  079A C30004             JMP   START
                ;
  079D 2A0510   ERRIN:    LHLD  NCNTR
  07A0 F9                 SPHL
  07A1 E1                 POP   H
  07A2 220110             SHLD  COBJ
  07A5 D1                 POP   D
  07A6 D1                 POP   D
  07A7 D5       INPUT:    PUSH  D
  07A8 CD7E0A             CALL  PR10
  07AB C3B707             JMP   INPT2
                ;
  07AE CD3404   INPT1:    CALL  TSTV
  07B1 DAF507             JC    INPT6
  07B4 C3C907             JMP   INPT4
  07B7 D5       INPT2:    PUSH  D
  07B8 CD3404             CALL  TSTV
  07BB DACC09             JC    LERMS
  07BE 1A                 LDAX  D
  07BF 4F                 MOV   C,A
  07C0 97                 SUB   A
  07C1 12                 STAX  D
  07C2 D1                 POP   D
  07C3 CD700A             CALL  MSG
  07C6 79                 MOV   A,C
  07C7 1B                 DCX   D
  07C8 12                 STAX  D
                ;
  07C9 D5       INPT4:    PUSH  D
  07CA EB                 XCHG
  07CB 2A0110             LHLD  COBJ
  07CE E5                 PUSH  H
  07CF 21A707             LXI   H,INPUT
  07D2 220110             SHLD  COBJ
  07D5 210000             LXI   H,0000H
  07D8 39                 DAD   SP
  07D9 220510             SHLD  NCNTR
  07DC D5                 PUSH  D
  07DD 3E3A               MVI   A,3AH
  07DF CD020A             CALL  GETL
  07E2 113713             LXI   D,LBUF
  07E5 CD1904             CALL  EEXPR
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                 PAGE 14


  07E8 00                 NOP
  07E9 00                 NOP
  07EA 00                 NOP
  07EB D1                 POP   D
  07EC EB                 XCHG
  07ED 73                 MOV   M,E
  07EE 23                 INX   H
  07EF 72                 MOV   M,D
  07F0 E1                 POP   H
  07F1 220110             SHLD  COBJ
  07F4 D1                 POP   D
                ;
  07F5 F1       INPT6:    POP   PSW
                +         TST   ''',''',INPT7
  07F6 CD0704   +         CALL  TEST
  07F9 2C       +         DB    ','
  07FA 03       +         DB    007FEH-$-1

  07FB C3A707             JMP   INPUT
  07FE C32E04   INPT7:    JMP   ENDL
                ;
  0801 1A       CMDER:    LDAX  D
  0802 FE0D               CPI   0DH
  0804 CA1208             JZ    LTEND
                ;
  0807 CD9A09   LET:      CALL  LTSUB
                +         TST   ''',''',LTEND
  080A CD0704   +         CALL  TEST
  080D 2C       +         DB    ','
  080E 03       +         DB    00812H-$-1

  080F C30708             JMP   LET
  0812 C32E04   LTEND:    JMP   ENDL
                ;
  0815 21A805   RST31:    LXI   H,ROPKW-1
  0818 C3C205             JMP   NXTKW
                ;
  081B CD4408   GE:       CALL  IFEXQ
  081E D8                 RC
  081F 6F                 MOV   L,A
  0820 C9                 RET
                ;
  0821 CD4408   NE:       CALL  IFEXQ
  0824 C8                 RZ
  0825 6F                 MOV   L,A
  0826 C9                 RET
                ;
  0827 CD4408   GT:       CALL  IFEXQ
  082A C8                 RZ
  082B D8                 RC
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1              PAGE 15

082C 6F               MOV   L,A
082D C9               RET
                 ;
082E CD4408   LE:     CALL  IFEXQ
0831 6F               MOV   L,A
0832 C8               RZ
0833 D8               RC
0834 6C               MOV   L,H
0835 C9               RET
                 ;
0836 CD4408   EQ:     CALL  IFEXQ
0839 C0               RNZ
083A 6F               MOV   L,A
083B C9               RET
                 ;
083C CD4408   LT:     CALL  IFEXQ
083F D0               RNC
0840 6F               MOV   L,A
0841 C9               RET
                 ;
0842 E1       NOROP:  POP   H
0843 C9               RET
                 ;
0844 79       IFEXQ:  MOV   A,C           ;SAVE C
0845 E1               POP   H             ;EXCHANGE STACK TOP 2
0846 C1               POP   B
0847 E5               PUSH  H
0848 C5               PUSH  B
0849 4F               MOV   C,A           ;RECOVER C
084A CD5908           CALL  EXPR
084D EB               XCHG                ;EXCHANGE LP AND 1 AND 2 EXPR
084E E3               XTHL
084F CD9009           CALL  CMINT
0852 D1               POP   D
0853 210000           LXI   H,0000H
0856 3E01             MVI   A,01H
0858 C9               RET                 ;SET C AND Z FF
                 ;
             +EXPR:   TST   '''-''',EXPR1
0859 CD0704   +       CALL  TEST
085C 2D       +       DB    '-'
085D 06       +       DB    00864H-$-1

085E 210000           LXI   H,0000H
0861 C38B08           JMP   NEGA1
                 ;
             +EXPR1:  TST   '''+''',EXPR3
0864 CD0704   +       CALL  TEST
0867 2B       +       DB    '+'
0868 00       +       DB    00869H-$-1
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1              PAGE 16

0869 CD9508   EXPR3:  CALL  TERM
                 ;
             +EXPR2:  TST   '''+''',NEGA0
086C CD0704   +       CALL  TEST
086F 2B       +       DB    '+'
0870 15       +       DB    00886H-$-1

0871 E5               PUSH  H             ;SAVE VALUE
0872 CD9508           CALL  TERM
0875 EB       ADDDBL: XCHG
0876 E3               XTHL                ;EXCHANGE LP AND 1 AND 2 VALUE
0877 7C               MOV   A,H
0878 AA               XRA   D
0879 7A               MOV   A,D
087A 19               DAD   D
087B D1               POP   D             ;RECOVER LINE POINTER
087C FA6C08           JM    EXPR2         ;CHECK OVERFLOW
087F AC               XRA   H
0880 F26C08           JP    EXPR2
0883 C3A104           JMP   ERROR

             +NEGA0:  TST   '''-''',EXPRT
0886 CD0704   +       CALL  TEST
0889 2D       +       DB    '-'
088A 8F       +       DB    0091AH-$-1
                 ;
088B E5       NEGA1:  PUSH  H             ;SAVE VALUE
088C CD9508           CALL  TERM
088F CD8409           CALL  TWSCP
0892 C37508           JMP   ADDDBL
                 ;
0895 CDF608   TERM:   CALL  FACTR
                 ;
             +MULT:   TST   '''*''',DIV
0898 CD0704   +       CALL  TEST
089B 2A       +       DB    '*'
089C 2C       +       DB    008C9H-$-1

089D E5               PUSH  H             ;SAVE FACTOR VALUE
089E CDF608           CALL  FACTR
08A1 0600             MVI   B,00H         ;SET SIGN PLUS
08A3 CD8109           CALL  ABS
08A6 EB               XCHG
08A7 E3               XTHL
08A8 CD8109           CALL  ABS
08AB 7C               MOV   A,H
08AC B7               ORA   A
08AD CAB608           JZ    MULT1
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1              PAGE 17

08B0 7A               MOV   A,D
08B1 B2               ORA   D
08B2 EB               XCHG
08B3 C2A204           JNZ   ERR1          ;IF BOTH MULTIPLIER>#00FF OVERFLOW
08B6 7D       MULT1:  MOV   A,L
08B7 210000           LXI   H,0000H
08BA B7               ORA   A
08BB CAE808           JZ    TERM1         ;IF ONE OF MULTIPLIER=0 VALUE=0
08BE 19       MULTL:  DAD   D
08BF DAA204           JC    ERR1          ;OVERFLOW?
08C2 3D               DCR   A
08C3 C2BE08           JNZ   MULTL
08C6 C3E808           JMP   TERM1
                 ;
             +DIV:    TST   '''/''',EXPRT
08C9 CD0704   +       CALL  TEST
08CC 2F       +       DB    '/'
08CD 4C       +       DB    0091AH-$-1

08CE E5               PUSH  H             ;SAVE VALUE
08CF CDF608           CALL  FACTR
08D2 0600             MVI   B,00H         ;SET SIGN PLUS
08D4 CD8109           CALL  ABS
08D7 EB               XCHG
08D8 E3               XTHL
08D9 CD8109           CALL  ABS
08DC 7A               MOV   A,D
08DD B3               ORA   E
08DE CAA204           JZ    ERR1          ;ZERO DIV?
08E1 C5               PUSH  B             ;SAVE SIGN
08E2 CD6409           CALL  DIVID
08E5 60               MOV   H,B
08E6 69               MOV   L,C
08E7 C1               POP   B             ;RECOVER SIGN
08E8 D1       TERM1:  POP   D
08E9 7C               MOV   A,H
08EA B7               ORA   A
08EB FAA104           JM    ERROR         ;OVERFLOW?
08EE 78               MOV   A,B
08EF B7               ORA   A
08F0 FC8409           CM    TWSCP
08F3 C39808           JMP   MULT
                 ;
08F6 218805   FACTR:  LXI   H,FNKW-1
08F9 C3C205           JMP   NXTKW
                 ;
08FC CD3404   FACT2:  CALL  TSTV          ;FROM KEY WORD MISMATCH
08FF DA0709           JC    FNUM
0902 7E               MOV   A,M
0903 23               INX   H
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1              PAGE 18

0904 66               MOV   H,M
0905 6F               MOV   L,A
0906 C9               RET                 ;RETURN VALUE ON HL
                 ;
0907 CD7704   FNUM:   CALL  GINT
090A 78               MOV   A,B
090B B7               ORA   A
090C C0               RNZ
                 ;
             +EXPRO:  TST   '''(''',EXPER
090D CD0704   +       CALL  TEST
0910 28       +       DB    '('
0911 09       +       DB    0091BH-$-1

0912 CD1904           CALL  EEXPR
              +       TST   ''')''',EXPER
0915 CD0704   +       CALL  TEST
0918 29       +       DB    ')'
0919 01       +       DB    0091BH-$-1

091A C9       EXPRT:  RET
091B C3CC09   EXPER:  JMP   LERMS
                 ;
091E CD0D09   RND:    CALL  EXPRO
0921 7C               MOV   A,H
0922 B7               ORA   A
0923 FAA104           JM    ERROR
0926 B5               ORA   L
0927 CAA104           JZ    ERROR
092A D5               PUSH  D
092B E5               PUSH  H
092C 2A1110           LHLD  RWRK
092F 11FF07           LXI   D,07FFH
0932 CD2004           CALL  COMP
0935 DA3B09           JC    RND1
0938 210000           LXI   H,0000H
093B 5E       RND1:   MOV   E,M
093C 23               INX   H
093D 56               MOV   D,M
093E 221110           SHLD  RWRK
0941 E1               POP   H
0942 EB               XCHG
0943 C5               PUSH  B
0944 CD6409           CALL  DIVID
0947 C1               POP   B
0948 D1               POP   D
0949 23               INX   H
094A C9               RET
                 ;
094B CD0D09   FNABS:  CALL  EXPRO
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 19


  094E CD8109            CALL  ABS
  0951 7C                MOV   A,H
  0952 B4                ORA   H
  0953 FAA104            JM    ERROR
  0956 C9                RET

  0957 2A1310   RSIZE:   LHLD  OBTM            ;VTOP-(CURRENTLB)
  095A D5                PUSH  D
  095B EB                XCHG
  095C 210013            LXI   H,VTOP
  095F CD7A09            CALL  DIFF
  0962 D1                POP   D
  0963 C9                RET

  0964 E5       DIVID:   PUSH  H
  0965 6C                MOV   L,H
  0966 2600              MVI   H,00H
  0968 CD6F09            CALL  DIVSB
  096B 41                MOV   B,C
  096C 7D                MOV   A,L
  096D E1                POP   H
  096E 67                MOV   H,A
  096F 0EFF     DIVSB:   MVI   C,0FFH
  0971 0C       DIVS1:   INR   C
  0972 CD7A09            CALL  DIFF
  0975 D27109            JNC   DIVS1
  0978 19                DAD   D
  0979 C9                RET

  097A 7D       DIFF:    MOV   A,L             ;HL_HL-DE
  097B 93                SUB   E
  097C 6F                MOV   L,A
  097D 7C                MOV   A,H
  097E 9A                SBB   D
  097F 67                MOV   H,A
  0980 C9                RET

  0981 7C       ABS:     MOV   A,H             ;ABSOLUTE->HL SIGN->B
  0982 B7                ORA   A
  0983 F0                RP
  0984 7C       TWSCP:   MOV   A,H             ;HL_-HL
  0985 2F                CMA
  0986 67                MOV   H,A
  0987 7D                MOV   A,L
  0988 2F                CMA
  0989 6F                MOV   L,A
  098A 23                INX   H
  098B 78                MOV   A,B             ;SIGN EXCHANGE
  098C EE80              XRI   80H
  098E 47                MOV   B,A
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 20


  098F C9                RET

  0990 7C       CMINT:   MOV   A,H             ;SIGN CHECK
  0991 AA                XRA   D
  0992 F29609            JP    CPIN1
  0995 EB                XCHG
  0996 CD2004   CPIN1:   CALL  COMP            ;COMPARE
  0999 C9                RET

  099A CD3404   LTSUB:   CALL  TSTV
  099D DACC09            JC    LERMS
  09A0 E5                PUSH  H
                +        TST   '''=''',LTERR
  09A1 CD0704   +        CALL  TEST
  09A4 3D       +        DB    '='
  09A5 0A       +        DB    009B0H-$-1

  09A6 CD1904            CALL  EEXPR
  09A9 44                MOV   B,H
  09AA 4D                MOV   C,L
  09AB E1                POP   H
  09AC 71                MOV   M,C
  09AD 23                INX   H
  09AE 70                MOV   M,B
  09AF C9                RET
  09B0 C3CC09   LTERR:   JMP   LERMS

               +TSTSC:   TST   ''';''',TSCR1
  09B3 CD0704   +        CALL  TEST
  09B6 3B       +        DB    ';'
  09B7 04       +        DB    009BCH-$-1

  09B8 F1                POP   PSW
  09B9 C31706            JMP   NXTGO
                ;
               +TSCR1:   TST   0DH,TSRTN
  09BC CD0704   +        CALL  TEST
  09BF 0D       +        DB    0000DH
  09C0 04       +        DB    009C5H-$-1

  09C1 F1                POP   PSW
  09C2 C30706            JMP   RUN1
  09C5 C9       TSRTN:   RET

  09C6 CD2604   TSCR2:   CALL  SKIP
  09C9 FE0D              CPI   0DH
  09CB C8                RZ

  09CC D5       LERMS:   PUSH  D               ;SAVE LINE POINTER
  09CD 11B004   LEMS0:   LXI   D,WHTMS
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 21


09D0 97       LEMS1:   SUB   A
09D1 CD700A            CALL  MSG
09D4 D1                POP   D               ;RECOVER LINE POINTER
09D5 1A                LDAX  D
09D6 F5                PUSH  PSW
09D7 97                SUB   A
09D8 12                STAX  D
09D9 2A0110            LHLD  COBJ
09DC E5                PUSH  H
09DD 7E                MOV   A,M
09DE 23                INX   H
09DF B6                ORA   M
09E0 D1                POP   D
09E1 CA0004            JZ    START
09E4 7E                MOV   A,M
09E5 B7                ORA   A
09E6 FA9D07            JM    ERRIN
09E9 CDF00A            CALL  WLINE
09EC 1B                DCX   D
09ED F1                POP   PSW
09EE 12                STAX  D
09EF 3E3F              MVI   A,3FH           ;'?'
09F1 CD1104            CALL  PUT
09F4 97                SUB   A
09F5 CD700A            CALL  MSG
09F8 C30004            JMP   START

09FB D5       SYSOF:   PUSH  D
09FC 11B604   SYS01:   LXI   D,SRYMS
09FF C3D009            JMP   LEMS1

0A02 CD1104   GETL:.   CALL  PUT
0A05 113713            LXI   D,LBUF
0A08 CD7E0B   GETL1:   CALL  BREAK           ;GET CHARACTER
0A0B CA080A            JZ    GETL1
0A0E CD1104            CALL  PUT             ;ECHO
0A11 FE0A              CPI   0AH
0A13 CA080A            JZ    GETL1           ;NEG. LF
0A16 B7                ORA   A
0A17 CA080A            JZ    GETL1           ;NEG. NUL
0A1A FE7F              CPI   7FH
0A1C CA2F0A            JZ    RUB             ;RUB?
0A1F FE7D              CPI   7DH             ; ?
0A21 CA3E0A            JZ    CAN
0A24 12                STAX  D               ;STORE INTO LINE BUFFER
0A25 13                INX   D
0A26 FE0D              CPI   0DH             ;CR?
0A28 C8                RZ
0A29 7B                MOV   A,E             ;BUF. FULL?
0A2A FE7F              CPI   7FH
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                    PAGE 22


0A2C C2080A            JNZ   GETL1

0A2F 7B       RUB:     MOV   A,E             ;BUF EMPTY?
0A30 FE37              CPI   37H
0A32 CA3E0A            JZ    CAN
0A35 1B                DCX   D
0A36 3E5C              MVI   A,5CH           ;"¥" PRINT
0A38 CD1104            CALL  PUT
0A3B C3080A            JMP   GETL1

0A3E CD0F04   CAN:     CALL  CRLF            ;CANCEL LINE
0A41 3E5E              MVI   A,5EH
0A43 C3020A            JMP   GETL

0A46 7C       SRCH:    MOV   A,H             ;SEARCH CURRENT LINE POSITION
0A47 B7                ORA   A               ;HL=INPUT LABEL NO
0A48 FAA104            JM    ERROR
0A4B 111510            LXI   D,OTOP
0A4E E5       SRCH1:   PUSH  H
0A4F 2A1310            LHLD  OBTM
0A52 2B                DCX   H
0A53 CD2004            CALL  COMP            ;COMPARE OBJ BOTTOM TO OBJ TOP OR CURRENT LINE
0A56 E1                POP   H
0A57 D8                RC
0A58 1A                LDAX  D               ;COMPARE OBJ LABEL TO INPUT LABEL
0A59 95                SUB   L
0A5A 47                MOV   B,A
0A5B 13                INX   D
0A5C 1A                LDAX  D
0A5D 9C                SBB   H
0A5E DA650A            JC    SKPL1           ;IF < SKIP LINE
0A61 1B                DCX   D
0A62 B0                ORA   B
0A63 C9                RET                   ;RETURN GT(A<>0) EQ(A=0)

0A64 13       SKIPL:   INX   D
0A65 13       SKPL1:   INX   D
0A66 1A       SKPL2:   LDAX  D
0A67 FE0D              CPI   0DH
0A69 C2650A            JNZ   SKPL1           ;SKIP UNTIL 'CR'
0A6C 13                INX   D
0A6D C34E0A            JMP   SRCH1

0A70 47       MSG:     MOV   B,A
0A71 1A       MSG1:    LDAX  D
0A72 13                INX   D
0A73 B8                CMP   B
0A74 C8                RZ
0A75 CD1104            CALL  PUT
0A78 FE0D              CPI   0DH
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                PAGE 23


 0A7A C2710A              JNZ    MSG1
 0A7D C9                  RET
                  ;
                +PR10:    TST    '''"''',PR13
 0A7E CD0704    +         CALL   TEST
 0A81 22        +         DB     '"'
 0A82 0F        +         DB     00A92H-$-1

 0A83 3E22                MVI    A,22H
 0A85 CD700A     PR11:    CALL   MSG
 0A88 FE0D                CPI    0DH
 0A8A E1                  POP    H
 0A8B CA0706              JZ     RUN1
 0A8E 23         PR12:    INX    H
 0A8F 23                  INX    H
 0A90 23                  INX    H
 0A91 E9                  PCHL
                  ;
                +PR13:    TST    ''''''''''',PR14
 0A92 CD0704    +         CALL   TEST
 0A95 27        +         DB     ''''
 0A96 05        +         DB     00A9CH-$-1

 0A97 3E27                MVI    A,27H
 0A99 C3850A              JMP    PR11
                  ;
                +PR14:    TST    '''_''',PR15
 0A9C CD0704    +         CALL   TEST
 0A9F 5F        +         DB     '_'
 0AA0 0C        +         DB     00AADH-$-1

 0AA1 3E8D                MVI    A,8DH
 0AA3 CD1104              CALL   PUT
 0AA6 CD1104              CALL   PUT
 0AA9 E1                  POP    H
 0AAA C38E0A              JMP    PR12
                  ;
 0AAD C9         PR15:    RET
                  ;
 0AAE D5         WINT:    PUSH   D                  ;WRITE INTEGER
 0AAF 110A00              LXI    D,000AH
 0AB2 D5                  PUSH   D
 0AB3 42                  MOV    B,D
 0AB4 0D                  DCR    C
 0AB5 CD8109              CALL   ABS
 0AB8 F2BE0A              JP     WINT1
 0ABB 062D                MVI    B,2DH
 0ABD 0D                  DCR    C
 0ABE C5         WINT1:   PUSH   B
 0ABF CD6409     WINT2:   CALL   DIVID
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                PAGE 24


 0AC2 78                  MOV    A,B
 0AC3 B1                  ORA    C
 0AC4 CACF0A              JZ     WINT3
 0AC7 E3                  XTHL
 0AC8 2D                  DCR    L
 0AC9 E5                  PUSH   H
 0ACA 60                  MOV    H,B
 0ACB 69                  MOV    L,C
 0ACC C3BF0A              JMP    WINT2
 0ACF C1         WINT3:   POP    B
 0AD0 0D         WINT4:   DCR    C
 0AD1 79                  MOV    A,C
 0AD2 B7                  ORA    A
 0AD3 FADE0A              JM     WINT5
 0AD6 3E20                MVI    A,20H
 0AD8 CD1104              CALL   PUT
 0ADB C3D00A              JMP    WINT4
 0ADE 78         WINT5:   MOV    A,B
 0ADF CD1104              CALL   PUT
 0AE2 5D                  MOV    E,L
 0AE3 7B         WINT6:   MOV    A,E
 0AE4 FE0A                CPI    0AH
 0AE6 D1                  POP    D
 0AE7 C8                  RZ
 0AE8 C630                ADI    30H
 0AEA CD1104              CALL   PUT
 0AED C3E30A              JMP    WINT6
                  ;
 0AF0 1A         WLINE:   LDAX   D                  ;HL_(DE),(DE+1)
 0AF1 6F                  MOV    L,A
 0AF2 13                  INX    D
 0AF3 1A                  LDAX   D
 0AF4 67                  MOV    H,A
 0AF5 13                  INX    D
 0AF6 0E04                MVI    C,04H              ;C=COLUMN
 0AF8 CDAE0A              CALL   WINT
 0AFB 3E20                MVI    A,20H
 0AFD CD1104              CALL   PUT
 0B00 97                  SUB    A
 0B01 CD700A              CALL   MSG
 0B04 C9                  RET
                  ;
 0B05 CD2004     TRNSF:   CALL   COMP               ;(BC) _(DE) (HL-1)
 0B08 C8                  RZ
 0B09 1A                  LDAX   D
 0B0A 02                  STAX   B
 0B0B 13                  INX    D
 0B0C 03                  INX    B
 0B0D C3050B              JMP    TRNSF
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                PAGE 25


 0B10 78         TR2:     MOV    A,B                ;(BC+1) (DE)_ (HL-1)
 0B11 92                  SUB    D
 0B12 C2180B              JNZ    TR2E
 0B15 79                  MOV    A,C
 0B16 93                  SUB    E
 0B17 C8                  RZ                        ;IF BC=DE RETURN
 0B18 1B         TR2E:    DCX    D
 0B19 2B                  DCX    H
 0B1A 1A                  LDAX   D
 0B1B 77                  MOV    M,A
 0B1C C3100B              JMP    TR2
                  ;
 0B1F C1         POPV:    POP    B
 0B20 E1                  POP    H
 0B21 220710              SHLD   FCNTR
 0B24 7C                  MOV    A,H
 0B25 B5                  ORA    L
 0B26 CA390B              JZ     NPOP
 0B29 E1                  POP    H
 0B2A 220910              SHLD   FSTPV
 0B2D E1                  POP    H
 0B2E 220B10              SHLD   FTOV
 0B31 E1                  POP    H
 0B32 220D10              SHLD   FLABL
 0B35 E1                  POP    H
 0B36 220F10              SHLD   FOBJ
 0B39 C5         NPOP:    PUSH   B
 0B3A C9                  RET
                  ;
 0B3B 21A713     PSHV:    LXI    H,MSTK
 0B3E CD8409              CALL   TWSCP
 0B41 C1                  POP    B
 0B42 39                  DAD    SP
 0B43 D2FB09              JNC    SYSOF
 0B46 2A0710              LHLD   FCNTR
 0B49 7C                  MOV    A,H
 0B4A B5                  ORA    L
 0B4B CA610B              JZ     NPSH
 0B4E 2A0F10              LHLD   FOBJ
 0B51 E5                  PUSH   H
 0B52 2A0D10              LHLD   FLABL
 0B55 E5                  PUSH   H
 0B56 2A0B10              LHLD   FTOV
 0B59 E5                  PUSH   H
 0B5A 2A0910              LHLD   FSTPV
 0B5D E5                  PUSH   H
 0B5E 2A0710              LHLD   FCNTR
 0B61 E5         NPSH:    PUSH   H
 0B62 C5                  PUSH   B
 0B63 C9                  RET
```

```
ISIS 8080 MACRO ASSEMBLER, V1.1                PAGE 26


 0B64 C2690B     TTYO1:   JNZ    TTYO0
 0B67 F1                  POP    PSW
 0B68 C9                  RET

 0B69 DBFB       TTYO0:   IN     0FBH
 0B6B E601                ANI    01H
 0B6D CA690B              JZ     TTYO0
 0B70 F1                  POP    PSW
 0B71 D3FA                OUT    0FAH
 0B73 FE0D                CPI    0DH
 0B75 C0                  RNZ
 0B76 3E0A                MVI    A,0AH
 0B78 CD1104              CALL   PUT
 0B7B 3E0D                MVI    A,0DH
 0B7D C9                  RET

 0B7E DBFB       BREAK:   IN     0FBH
 0B80 00                  NOP
 0B81 E602                ANI    02H
 0B83 C8                  RZ
 0B84 DBFA       TTYI1:   IN     0FAH
 0B86 E67F                ANI    7FH
 0B88 FE0F                CPI    0FH
 0B8A C2970B              JNZ    TTYI2
 0B8D 3A0010              LDA    OMASK              ;INVERSE MASK
 0B90 2F                  CMA
 0B91 320010              STA    OMASK
 0B94 C37E0B              JMP    BREAK

 0B97 FE03       TTYI2:   CPI    03H
 0B99 C0                  RNZ
 0B9A C30004              JMP    START


 1000                     ORG    VENT
 1000 FF         OMASK:   DB     0FFH
 1001            COBJ:    DS     0002H
 1003            RSTCK:   DS     0002H
 1005            NCNTR:   DS     0002H
 1007            FCNTR:   DS     0002H
 1009            FSTPV:   DS     0002H
 100B            FTOV:    DS     0002H
 100D            FLABL:   DS     0002H
 100F            FOBJ:    DS     0002H
 1011            RWRK:    DS     0002H
 1013 1510       OBTM:    DW     OTOP
 1015            OTOP:    DS     0001H
 0400                     END    START
```

東大版 8080 2K BASIC ②ダンプリスト

```
O BLOCK01 0
O ABS    00981H
O ADDDB  00875H
O BREAK  00B7EH
O CAN    00A3EH
O CASE   0F721H
O CMDER  00801H
O CMDKW  00535H
O CMINT  00990H
O COBJ   01001H
O COMP   00420H
O CPIN1  00996H
O CRLF   0040FH
O DIFF   0097AH
O DIV    008C9H
O DIVID  00964H
O DIVS1  00971H
O DIVSB  0096FH
O EEXPR  00419H
O ENDL   0042EH
O ENTRY  004BCH
O EQ     00836H
O ERR1   004A2H
O ERRIN  0079DH
O ERROR  004A1H
O EXEQT  005EAH
O EXPER  0091BH
O EXPR   00859H
O EXPRO  0090DH
O EXPR1  00864H
O EXPR2  0086CH
O EXPR3  00869H
O EXPRT  0091AH
O FACT2  008FCH
O FACTR  008F6H
O FCNTR  01007H
O FLABL  0100DH
O FNABS  0094BH
O FNKW   00589H
O FNUM   00907H
O FOBJ   0100FH
O FOR    006CCH
O FORO   006F1H
O FOR10  0072AH
O FOR3   00709H
O FORTO  006DCH
O FSTEP  006E8H
O FSTP1  006EEH
O FSTPV  01009H
O FTOV   0100BH
O GE     0081BH
O GETC   004D5H
O GETI1  0047EH
O GETL   00A02H
O GETL1  00A08H
O GINT   00477H
O GOSUB  0068FH
O GOTO   00620H
O GT     00827H
O HOWMS  004A8H
O IENT   00400H
O IFEXQ  00844H
O IFST2  00794H
O IFSTM  0078CH
O INPT1  007AEH
O INPT2  007B7H
O INPT4  007C9H
O INPT6  007F5H
O INPT7  007FEH
O INPUT  007A7H
O INSRT  004EAH
O KWC1   005C6H
O KWCPR  005BFH
O KWSK1  005D9H
O KWSK2  005E5H
O KWSRT  005E3H
O KWSTP  005A1H
O KWTO   0059BH
O LBUF   01337H
O LE     0082EH
O LEMSO  009CDH
O LEMS1  009D0H
O LERMS  009CCH
O LET    00807H
O LIST   00631H
O LISTL  0063AH
O LT     0083CH
O LTEND  00812H
O LTERR  009B0H
O LTSUB  0099AH
O MOVE   0050CH
O MSG    00A70H
O MSG1   00A71H
O MSTK   013A7H
O MULT   00898H
O MULT1  008B6H
O MULTL  008BEH
O NCNTR  01005H
O NE     00821H
O NEGAO  00886H
O NEGA1  0088BH
O NEW    005F2H
O NEXT   00731H
O NEXT1  0073AH
O NEXT2  00754H
O NEXT4  0076CH
O NEXT5  00780H
O NOROP  00842H
O NPOP   00B39H
O NPSH   00B61H
O NXTGO  00617H
O NXTKW  005C2H
O OBTM   01013H
O OKMES  004ADH
O OMASK  01000H
O OTOP   01015H
O POPV   00B1FH
O PR10   00A7EH
O PR11   00A85H
O PR12   00A8EH
O PR13   00A92H
O PR14   00A9CH
O PR15   00AADH
O PRINT  00649H
O PRNT1  00656H
O PRNT2  00661H
O PRNT3  0066DH
O PRNT4  00673H
O PRNT5  0067EH
O PRNT6  00684H
O PSHV   00B3BH
O PUT    00411H
O REM    00786H
O RETRN  006B1H
O RND    0091EH
O RND1   0093BH
O ROPKW  005A9H
O RSIZE  00957H
O RST31  00815H
O RSTCK  01003H
O RUB    00A2FH
O RUN    00601H
O RUN1   00607H
O RUN2   00610H
O RWRK   01011H
O SKIP   00426H
O SKIPL  00A64H
O SKPL1  00A65H
O SKPL2  00A66H
O SRCH   00A46H
O SRCH1  00A4EH
O SRYMS  004B6H
O ST2    004CCH
O STACK  01400H
O START  00400H
O STMKW  00545H
O STOP   005FBH
O SYS01  009FCH
O SYSOF  009FBH
O TERM   00895H
O TERM1  008E8H
O TEST   00407H
O TEST1  00468H
O TR2    00B10H
O TR2E   00B18H
O TRNSF  00B05H
O TSCR1  009BCH
O TSCR2  009C6H
O TSRTN  009C5H
O TST    0F775H
O TSTEQ  00473H
O TSTSC  009B3H
O TSTV   00434H
O TSTV1  00458H
O TTYI1  00B84H
O TTYI2  00B97H
O TTYOO  00B69H
O TTYO1  00B64H
O TWSCP  00984H
O VENT   01000H
O VTOP   01300H
O WHTMS  004B0H
O WINT   00AAEH
O WINT1  00ABEH
O WINT2  00ABFH
O WINT3  00ACFH
O WINT4  00AD0H
O WINT5  00ADEH
O WINT6  00AE3H
O WLINE  00AF0H
O BLOCK02 0
O BLOCK03 0
O BLOCK04 0
O BLOCK05 0
O BLOCK06 0
O BLOCK07 0
O BLOCK08 0
O BLOCK09 0
O BLOCK0A Q
O BLOCK0B 0
O BLOCK0C 0
O BLOCK0D 0
O BLOCK0E 0
O BLOCK0F 0
O BLOCK10 0
O BLOCK11 0
O BLOCK12 0
```

```
0 BLOCK13 0
0 BLOCK14 0
0 BLOCK15 0
0 BLOCK16 0
0 BLOCK17 0
0 BLOCK18 0
0 BLOCK19 0
0 BLOCK1A 0
0 BLOCK1B 0
0 BLOCK1C 0
0 BLOCK1D 0
0 BLOCK1E 0
0 BLOCK1F 0
0 BLOCK20 0
0 BLOCK21 0
0 BLOCK22 0
0 BLOCK23 0
0 BLOCK24 0
0 BLOCK25 0
0 BLOCK26 0
0 BLOCK27 0
0 BLOCK28 0
0 BLOCK29 0
0 BLOCK2A 0
0 BLOCK2B 0
0 BLOCK2C 0
0 BLOCK2D 0
0 BLOCK2E 0
```

↓ここから録音されています。

```
$
:10040000F3310014C3BC04E3CD2604BEC368043E2C
:100410000DF53A0010B7C3640BCD5908E5C31508B4
:100420007CBAC07DBBC91AFE20C013C32604CDB35D
:1004300009C3CC09CD2604D640D8C2580413CD0D2B
:10044000929DAA104D5EBCD5709CD2004DAFC093E
:1004500021001 3CD7A09D1C9FE1B3FD81321001307
:100460007856F3E008C67C923CA7304C54E06001A
:1004700009C11B1323E3C921000044CD2604FE302B
:10048000D8FE3AD03EF0A4C2A10404C5444D2929A7
:100490009291A13E60F856F3E008C67C11AF27E98
:1004A0004D511A804C3D009484F573F0D4F4B0D39
:1004B000574841543F0D534F5252590DCD0F04111F
:1004C000AD0497CD700A21CD04220110210000223 5
:1004D00071022031 03E3ECD020AD5113713CD7707
:1004E0004CD26047CB5C1CABF051B7C121B7D123E
:1004F000C5D57993F5CD460AD5C20C05D5CD640A8C
:10050000C12A1310CD050B6069221310C12A1310E4
:10051000F1E5FE03CA0004856F3E008C67110013ED
:10052000CD2004D2FB09221310D1CD100BD1E1CD87
:10053000050BC3D5044C495354863152554E8601A0
:100540004E455785F24E45585487314C45548807DF
:100550004946878C474F544F8620474F5355428614
:1005600008F52455455524E86B152454D8786464F5F
:10057000052 86CC494E50555487A75052494E548606
:10058000 4953544F5085FB8801524E44891E4142C5
:1005900053894B53495A45895788FC544F86DC8907
:1005A000CC5354455006E034CE3E3D881B23882177
:1005B0003E88273D88363C3D882E3C883C88422139
:1005C0003405CD2604D51A13FE2ECAE30523BECA70
:1005D000C6053E7F1BBEDAEA0523BED2D90523D16C
:1005E000C3C2053E7F23BED2E5057E236EE67F674C
:1005F000F1E9CDC6092115102213 10CDC609C3009B
:100600004CDC609111510210000CD4E0ADA0004F0
:10061000EB220110EB1313CD7E0B214405C3C20561
:10062000CD1904D5CDC609CD460AC2A204F1C31026
:100630006CD7704CDC609CD460ADA0004CDF00A0E
:10064000CD7E0BCD4E0AC33A060E06CD07043B06FF
:10065000CD0F04C31706CD07040D06CD0F04C30745
:100660006CD07042307CD19044DC37306CD7E0ABA
:10067000C38406CD07042C06CDB309C36106CD0F94
:1006800004C32E04CD1904C5CDAE0AC1C37306CD73
:100690003B0BCD1904D5CD460AC2A2042A0110E5B0
:1006A0002A0310E5210000220710392203 10C3108D
:1006B0006CDC6092A03107CB5CACC09F9E122038C
:1006C00010E1220110D1CD1F0BC32E04CD3B0BCD69
:1006D0009A092B2207102 19A05C3C205CD190422BD
:1006E000B1021A005C3C205CD1904C3F1062101D9
:1006F00002209102A0110220D10EB220F10010A0E
:1007000002A0710EB6068393E097E23B6CA2A0723
:100710007E2BBAC209077EBBC20907EB2100003954
:1007200044D210A0019CD100BF92A0F10EBC32EEE
:100730004CD3404DACC09220510D5EB2A07107C4D
:10074000B5CACD09CD2004CA5407D1CD1F0B2A0547
:1007500010C33A075E23562A0910E519EB2A071041
:100760007323722A0B10F1B7F26C07EBCD9009D10D
:10077000DA80072A0D102201102A0F10EBC32E0475
:10078000CD1F0BC32E0421000C39407CD19047C98
:10079000B5C21706CD660AD21006C300042A05109A
:1007A000F9E1220110D1D1D5CD7E0AC3B707CD34EE
:1007B0004DAF507C3C907D5CD3404DACC091A4FDA
:1007C0009712D1CD700A791B12D5EB2A0110E521C1
:1007D000A7072201102100003922051 0D53E3ACD8D
:1007E00020A113713CD1904000000D1EB732372F4
:1007F000E1220110D1F1CD07042C03C3A707C32EBA
:1008000041AFE0DCA1208CD9A09CD07042C03C3A1
:10081000708C32E0421A805C3C205CD4408D86F1C
:10082000C9CD4408C86FC9CD4408C8D86FC9CD44E4
:1008300086FC8D86CC9CD4408C06FC9CD4408D072
:100840006FC9E1C979E1C1E5C54FCD5908EBE3CDE9
:100850009009D12100003E01C9CD07042D062100D9
:1008600000C38B08CD07042B00CD9508CD07042BC2
:1008700015E5CD9508EBE37CAA7A19D1FA6C08ACA2
:10088000F26C08C3A104CD07042D8FE5CD9508CDEA
:1008900008409C37508CDF608CD07042A2CE5CDF6EA
:1008A000080600CD8109EBE3CD81097CB7CAB60803
:1008B0007AB2EBC2A2047D210000B7CAE80819DAB7
:1008C000A2043DC2BE08C3E808CD07042F4CE5CD05
:1008D000F6080600CD8109EBE3CD81097AB3CAA2FF
:1008E0004C5CD6409606 9C1D17CB7FAA10478B7A9
:1008F000FC8409C3980821880 5C3C205CD3404DAF5
:100900007097E23666FC9CD770478B7C0CD070489
:1009100028 09CD1904CD07042901C9C3CC09CD0D7F
:1009200097CB7FAA104B5CAA104D5E52A111011B2
:10093000FF07CD2004DA3B092100005E2356221177
:10094000 10E1EBC5CD6409C1D123C9CD0D09CD811D
:1009500097CB4FAA104C92A1310D5EB210013CDE8
:100960007A09D1C9E56C2600CD6F09417DE1670E9A
:10097000FF0CCD7A09D27109 19C97D936F7C9A67F2
:10098000C97CB7F07C2F677D2F6F2378EE8047C935
:100990007CAAF29609EBCD2004C9CD3404DACC0947
:1009A000E5CD07043D0ACD1904444DE1712370C91A
:1009B000C3CC09CD07043B04F1C31706CD07040DD2
:1009C0004F1C30706C9CD2604FE0DC8D511B00435
:1009D00097CD700AD11AF597122A0110E57E23B639
:1009E000D1CA00047EB7FA9D07CDF00A1BF1123E72
:1009F0003FCD110497CD700AC30004D511B604C3CE
:100A0000D009CD1104113713CD7E0BCA080ACD11C0
:100A10004FE0ACA080AB7CA080AFE7FCA2F0AFEDD
:100A2007DCA3E0A1213FE0DC87BFE7FC2080A7BF8
:100A3000FE37CA3E0A1B3E5CCD1104C3080ACD0F27
:100A40043E5EC3020A7CB7FAA104111510E52A20
:100A50013102BCD2004E1D81A9547131A9CDA65A0
:100A60000A1BB0C913131AFE0DC2650A13C34E0A3E
:100A700471A13B8C8CD1104FE0DC2710AC9CD07BB
:100A800004220F3E22CD700AFE0DE1CA0706232381
:100A90023E9CD070427053E27C3850ACD07045F58
:100AA0000C3E8DCD1104CD1104E1C38E0AC9D511C0
:100AB0000A00D5420DCD8109F2BE0A062D0DC5CD25
:100AC000640978B1CACF0AE32DE56069C3BF0AC1E2
:100AD0000D79B7FADE0A3E20CD1104C3D00A78CDD5
:100AE00011045D7BFE0AD1C8C630CD1104C3E30AF0
:100AF0001A6F131A67130E04CDAE0A3E20CD1104EF
:100B000097CD700AC9CD2004C81A021303C3050B80
:100B10007892C2180B7993C81B2B1A77C3100BC19C
:100B2000E12207107CB5CA390BE1220910E1220B42
:100B30010E1220D10E1220F10C5C921A713CD84A9
:100B40009C139D2FB092A07107CB5CA610B2A0FEB
:100B50010E52A0D10E52A0B10E52A0910E52A07F1
:100B60010E5C5C9C2690BF1C9DBFBE601CA690B17
:100B7000F1D3FAFE0DC03E0ACD11043E0DC9DBFBD8
:100B800000E602C8DBFAE67FFE0FC2970B3A0010C0
:0D0B90002F320010C37E0BFE03C0C3000413
:01100000FFF0
:021013001510B6
:00040001FB
```

丸善洋書売場案内

●ソ連の科学・技術研究機関ガイド／第2版
Guide to Science and Technology in the USSR. 2nd Edition. 1977.
(F. Hodgson, London／日本総代理店：丸善)
〈近着〉……………………………… 予定価 ¥30,000

●アジア・太平洋諸国の科学・技術研究機関ガイド
Guide to Science and Technology in the Asia/Pacific Area. 1977.
(F. Hodegson, London/日本総代理店：丸善)
〈近着〉…………………………………予定価 ¥30,000

●イギリスの科学・技術研究機関ガイド／第2版
Guide to Science and Technology in the UK. 2nd Edition. 1977.
(F. Hodgson, London/日本総代理店：丸善)
〈近着〉…………………………………予定価 ¥30,000

●国際科学・技術研究機関ガイド
Guide to International Science and Technology. 1977. (F. Hodgson, London/日本総代理店：丸善)
〈近着〉…………………………………予定価 ¥30,000

《お問い合せ先》☎03(272)7211

BOOK GUIDE

工業調査会 ¥1,300

マイコンキット活用ブック

安田寿明編著

マイコン関係の類書が続々刊行される中で,『電子材料』,『自動化技術』誌などでおなじみの工業調査会から，手づくりコンピュータの手引書が出ました．

内容はその名の通りマイコン《キット》の使い方を示したもので，調査会らしく，キット・メーカー各社をもうらしています．

キットを数多くとりあげたのが本書の特長でしょう．

つまり今どんなキットがあって，どの程度のことをすれば，どの程度のものができるかの目安にはなります．あらゆるキットにとりくんでいるせいか詳しく記されていないのが残念ですが，まったくの入門者にとって，本書は非常に役立つと思います．

各社のカタログ，取説などがそれぞれ，一部分でも ¥1,300 で手に入るというのも魅力のひとつといえます．

でんごんばん

ムーンベース

外国製高級工具 Ungar半田ゴテセット，半田ゴテクリップ ¥ 1,690，ヒーター27W～50W ¥ 1,730～¥ 2,310，コテ先各種 ¥450～¥550

MPKタネモリ

I/O読者に部品のレイアウトや基板パターン下書にとても重宝する2.54ミリピッチのグラフ用紙（小さくて50×35目）を1人につき10枚差し上げます．
(〒733 広島市西十日市町2－1－403)
☎ (0822) 91－3659

斉藤電気

斉藤電気商会では，小型のパルスモーターを¥ 200で売っています．JAPAN SERVO製，PHASE; 4，STEP; 48，COIL; 500Ω，24Vと書いてありました．
(K²)

斉藤電気商会 東京都千代田区外神田1－10－11
(東京ラジオデパート内)

共立電子

共立電子ではI/Oの読者サービス (1,000円以上買いものをして，I/O誌の共立の広告を店員さんに示せばBA609を16桁分もらえるというもの) を通信販売にも拡大することになりました．このサービスは7月8月とつづいて来ましたが，各地のI/O読者よりの『通信販売にもサービスを適用すべきだ』との声に答えたもの．通販で1,000円以上買いものをした人はサービス券(共立の広告右下にある)を切り取っていっしょに送るべし!!

なお，このサービスは店頭，通販とも9月いっぱいで打切られ，日を新らたにして読者サービスをするとのことです．
☎ (06) 631－5936

データ プロ

大阪のデータープロがチビコンを発売してOEM間ではかなりの反響でしたが，この度一般アマチュアにもと言うことで日本橋の共立電子産業でもOEM価格で発売することになりました．
データプロ ☎ (06) 395－1571

■次号予告

9月25日発売の次号では，マイコンの鉄道模型への応用，MIL記号の使い方，DVM用LSIの使い方などの他，APPLE，PETなどで注目される6502について解説します．また，マイコンの周辺装置についての製作記事も掲載します．ご期待ください！

■編集後記

▶今月のI/Oはいかがでしたか？先月号から，色々新しい試みをしていますが，読者諸氏の感想はいかがでしょうか．ぜひ編集部へお聞かせください．

▶マイコンブームも全国的になり，東京・松屋などにつづいて札幌でもデパートでマイコンショーが開かれました．お客の動員数はいま一歩というところでしたが，将来マイコン・ファンの層の拡大を考えると意義のあるこころみだったと思います．

▶I/Oの発行部数も読者諸氏のご支援のおかげで順調に伸びており，マイコン・ブームもいよいよ本格化してきたといえそうです．

■バックナンバーのお知らせ

No.1（'76年11月号）〜No.7（'77年5月号）まではすべて品切れです．No.1〜No.4は合本1（¥1,900 送料160円）No.5〜No.7は合本2（¥1,900 送料160円）に収められています．

コピー・サービスもしていますが高価になってしまうので，なるべく合本をご利用ください．

なお，コピーサービスの価格は，

No.1が¥600，No.2が¥780，の他は全て¥960です．〒は2部まで120円，4部まで160円，5部200円です．

◘原稿募集

「I/O」はみんなの広場です．以下の各原稿を募集していますので，ぜひあなたも参加して下さい．

①イベント，ミーティング，講習会，勉強会 etc のお知らせ．

②製作・実験のレポート　原稿用紙（400字詰）3枚くらいにまとめる．図，表はエンピツ書きでOK．写真もぜひ入れて下さい．

③「I/Oポート」のマイコン・クラブ紹介（メンバーの写真も！）

④秋葉原の情報（お買徳品の情報 etc.）

⑤ソフトウェア道場　プログラムの説明とアセンブラまたはマシン語のリスト．フローチャートも．

②〜⑤は採用の場合には稿料をさしあげます．

なお，投稿の際には以下のことを必らず記入して下さい．

(イ)現在の所属（ペンネームの場合でも一応ご記入願います．）

(ロ)連絡先（勤務先または自宅）の住所，電話番号．

(ハ)年齢，学年

(ニ)現在所有しているマイコンがあればその名称（例：8080，6800，SC/MP）

編集部に対するご意見がありましたら，あわせて，お寄せ下さい．

■投稿先

〒151東京都渋谷区代々木2-5-1羽田ビル403 工学社内
日本マイクロコンピュータ連盟「投稿係」

◘定期講読のおすすめ

「I/O」は予約購読を原則とします．予約申し込みは半年，1年で，半年以上申し込まれた方は，「マイコン連盟」の会員として登録されます．

①1冊400円（送料込）
②半年…2,200円（送料込）
③1年…4,000円（送料込）

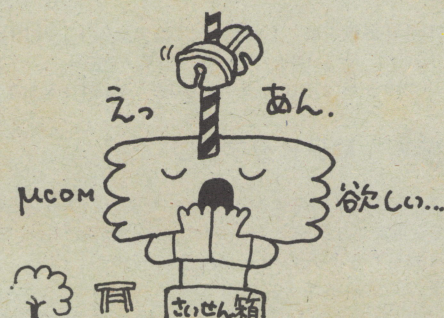
■団体割引
なお，5名以上で1年間の予約をする場合は団体会員として，1名当り年間3,500円をお支払い下さい．

■送付方法

①郵便振替《東京2−49427》
裏の通信欄に，何月号からご希望か明記してください．

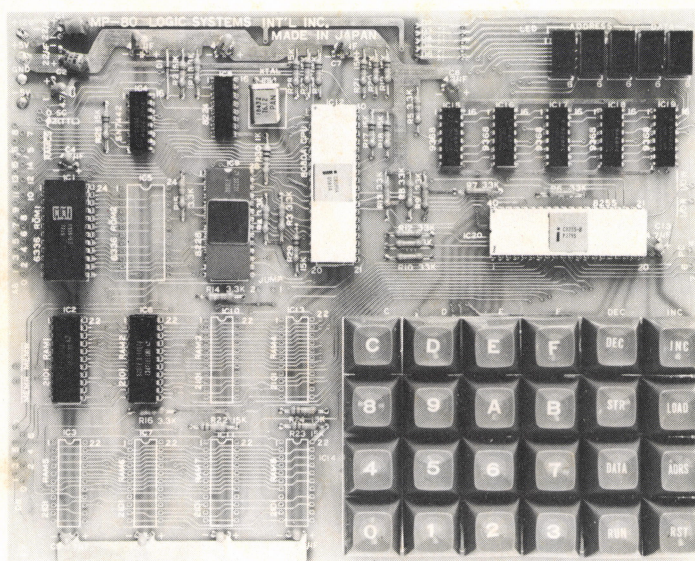
②現金書留
③定格小為替
のいずれか．
何月号からご希望か明記したものを，同封してください．

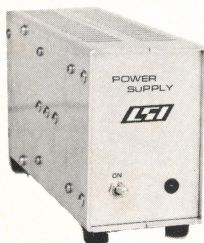
■送付先

〒151東京都渋谷区代々木2-5-1羽田ビル403 工学社内
「日本マイクロコンピュータ連盟」

月刊 I/O　1977年9月号　第2巻第9号（通巻第11号）
発行人　星　正明
編集人　森　昭助
編　集　日本マイクロコンピュータ連盟
発行所　株式会社　工学社
〒151 東京都渋谷区代々木2-5-1 羽田ビル403 ☎(03)375−5784 振替口座東京5-22510
印刷：耕文社

遂に出現３万円台の本格的マイコンキット

新発売

8080A(インテル社または相当品)使用
拡張が簡単
部品はすべて産業用高級品使用

マイコンキット　MP-80

定価　39,500円

▶部品点数が少なくて、だれにでも組立が簡単
▶親切な組立て説明書つき
▶専門のスタッフによる無料指導

主な仕様

CPU：8080A使用
RAM：256バイト（基板上1Kバイトまで拡張可）
ROM：256バイト（　〃512　〃　　　）
I/O：プログラムI/Oポート　8255A使用
入力：キーボード　24個
出力：16進表示　LEDディスプレイ５桁
電源：+5V 1.5A　+12V300mA　−5V 50mA以下

専用電源(別売)
定価　13,000円

Logic Systems International, Inc
ロジックシステムズ　インターナショナル株式会社
〒108 東京都港区三田４丁目15-29-342　電話(03)452-4994, 2514

取次先　株式会社　マイテック　（マイコン通信講座用教材）
〒103 東京都中央区日本橋茅場町2-1市川ビル 電話(03)-661-3366(代)

I/O
昭和52年9月号
第2巻第9号 通巻11号 昭和52年9月1日発行（毎月1回1日発行）
昭和52年1月11日 第三種郵便物認可

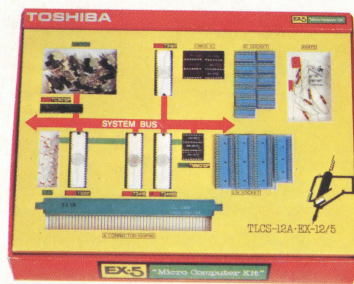
このボードから、マイコン・ライフが始まります。

新発売!!

手作りマイコン・キットTLCS-12A EX-12/5は、数時間でだれにでも組立てられる完全部品キットです。
マイコンを自分の手で作るという楽しみも、もちろんありますが、このEX-12/5には、組立後にアイデアを生かして限りなくシステムを発展できるという楽しみがあります。
EX-12/5で、あなたもマイコン・ライフをはじめませんか。

〈応用例〉
競馬ゲーム/ディジタルクロック/電子オルゴール/TTY接続/電光表示板/電子ルーレット/電子スロットマシン/モールスコンバータ/オーディオカセット接続/ビデオゲーム/OEM組込用etc.

標準価格77,000円

東芝ワンボード・マイクロコンピュータ・キット――TLCS-12A・EX-12/5

"Micro Computer Kit"

東芝

お問合せは　東京芝浦電気株式会社半導体事業部マイクロコンピュータ営業企画部　〒210 川崎市幸区堀川町72　TEL（044）522-2111（大代）

特価　350円

I/O

1977 9

■特集BASICを使いこなそう！【付録レコード】

●編集＝日本マイクロコンピュータ連盟

VOL 2 No. 9

工学社